Real Time Optimization of Chemical Processes — A Comparison of Closed Versus Open Form Equations Using Commercial Simulation Packages

by

Matthew R. Naysmith

A thesis
presented to the University of Waterloo
in fulfilment of the
thesis requirement for the degree of
Doctor of Philosophy
in
Chemical Engineering

Waterloo, Ontario, Canada, 1997

©Matthew R. Naysmith, 1997

National Library of Canada
Acquisitions and Bibliographic Services
395 Wellington Street
Ottawa ON K1A 0N4
Canada

Bibliothèque nationale du Canada
Acquisitions et services bibliographiques
395, rue Wellington
Ottawa ON K1A 0N4
Canada

Your file Votre référence

Our file Notre référence

The author has granted a non-exclusive licence allowing the National Library of Canada to reproduce, loan, distribute or sell copies of this thesis in microform, paper or electronic formats.

The author retains ownership of the copyright in this thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without the author's permission.

L'auteur a accordé une licence non exclusive permettant à la Bibliothèque nationale du Canada de reproduire, prêter, distribuer ou vendre des copies de cette thèse sous la forme de microfiche/film, de reproduction sur papier ou sur format électronique.

L'auteur conserve la propriété du droit d'auteur qui protège cette thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.

0-612-30635-6

Canada

The University of Waterloo requires the signatures of all persons using or photocopying this thesis. Please sign below, and give address and date.

Real Time Optimization of Chemical Processes — A Comparison of Closed Versus Open Form Equations Using Commercial Simulation Packages

Abstract

Real time optimization (RTO) is the continuous reevaluation and alteration of operating conditions of a process so that the economic productivity of the process is maximized subject to operational constraints. Current industrial real time optimizers are generally designed and coded manually for each application, using a programming language such as FORTRAN, PASCAL or C, in order to solve the optimization problem within a reasonable period of time. Recently, computers have become powerful enough to allow the use of chemical process simulation packages in real time optimization. Process simulation packages have the advantages of providing user friendly interfaces, a library of pre-built and tested unit operation models, rigorous property libraries and built in optimization algorithms. Therefore, real time optimization applications can be developed more easily and quickly, and are easier to maintain than those developed by direct coding.

The main objective of this research was to compare the use of closed and open form equation based process simulation packages as the process models in real time optimization. Currently, both closed form equation based, or sequential modular, and open form equation based, or equation-oriented, process simulators have been suggested for use in real time optimization, but not implemented.

Two real time optimizers were developed for an industrial refinery stabilizer-splitter process using ASPEN PLUS, a sequential modular process simulator, and SPEEDUP, an equation-oriented process simulator.

After testing the two real time optimizers, it was concluded that neither ASPEN PLUS nor SPEEDUP in their present form were robust enough for an industrial on-line application. The equation-oriented simulator, how-

ever, was able to find better objective function values using approximately half as many cycles of the real time optimizer than the sequential modular simulator. The sequential modular simulator had difficulty in converging the economic optimization problem when inequality constraints became violated. Problems were also encountered converging the tear stream blocks in ASPEN PLUS for some operating conditions. The SPEEDUP real time optimizer required weighting factors in the objective functions for both data reconciliation and economic optimization due to a lack of tuning parameters for the SRQP optimization algorithm and possible flowsheet variable scaling problems.

From the development of the two real time optimizers, it was concluded that sequential modular simulators are quicker and easier to use than equation oriented simulators. However, they are not as flexible with respect to the choice of manipulated variables for optimization and convergence of flowsheet recycle loops and tear streams. The development of the real time optimizers also provides a "test bed" for future research into real time optimization.

Overall, despite the added difficulty in developing and converging an equation-oriented process flowsheet model, the added flexibility in terms of variable specification, robust and rapid tear stream convergence, and more powerful optimization algorithms make the equation-oriented process simulator a more attractive tool for real time optimization, provided the problems encountered can be overcome. Equation oriented process simulation packages have also already been used in advanced model based control, scheduling and process operation analysis which has the advantage of providing the same user-interface and models for all process modelling performed.

Several other runs were performed with the ASPEN PLUS real time optimizer testing changes in the economic objective function parameters and the effect of process disturbances. Also, a sensitivity study of the economic objective function stream prices was performed which showed that in every case tested, the real time optimizer was able to find a better operating point after a change in economic parameters.

Acknowledgements

I would like to thank following people who assisted me in my research:

- Professor Peter Douglas for his encouragement and advice which were always welcome;
- Dr. Dave Radford of Shell Canada's Sarnia refinery for his considerable time spent helping me learn about real time optimization and the stabilizer-splitter process;
- Mr. Dennis Herman for his time in helping me solve my many computer and software problems;
- Mr. Raveen Sharma of Aspen Technology Inc. for his assistance with several process simulation and optimization software problems.

I would also like thank my family who have supported and encouraged me throughout my studies.

Finally, I would like thank my partner Natalie for her endless support and assistance over the last 4 years, without which I could not have achieved this work.

NOTE TO USERS

Page(s) missing in number only; text follows. Microfilmed as received.

x

UMI

Contents

1 Introduction 1

1.1 Objectives 1

1.2 Real Time Optimization 2

1.2.1 Real Time Optimization and Control Hierarchy 3

1.2.2 Current Real Time Optimization Systems 5

1.3 Process Simulation 6

1.3.1 Sequential Modular Process Simulators 8

1.3.2 Equation-oriented Process Simulators 10

1.4 Scope of this Research 11

1.5 Conventions Used in this Thesis 15

2 Literature Review 17

2.1 Components of a Real Time Optimizer 17

2.1.1 Steady State Detection 19

2.1.2 Data Reconciliation and Gross Error Detection 22

2.1.3 Parameter Estimation 33

2.1.4 Process Model 36

2.1.5 Optimization 39

2.2 Methods of Real Time Optimization 42

2.2.1 Global or Centralized Approach to Real Time Optimization 42

2.2.2 Distributed Approach to Real Time Optimization . . . 50

2.2.3 Direct Methods (Model free approaches) 55

2.3 Implementation 55

3 Process Simulation Model Development **59**

3.1 Introduction 59
3.1.1 Objectives 59
3.1.2 Process Description 59
3.1.3 Development Method 62
3.2 Properties 62
3.3 Sequential Modular Simulation 64
3.3.1 Stabilizer (C1) 64
3.3.2 Heat Exchanger (E1) 64
3.3.3 Splitter (C2, C4 and C5) 66
3.3.4 Convergence 66
3.4 Equation-oriented Simulation 67
3.4.1 User Defined Equipment Models 69
3.4.2 User Defined Measurement Models 69
3.4.3 Stabilizer (C1) 71
3.4.4 Heat Exchanger (E1) 72
3.4.5 Splitter (C2, C4 and C5) 73
3.4.6 Convergence 73
3.5 Discussion 75
3.5.1 Properties 75
3.5.2 Sequential Modular Simulation 77
3.5.3 Equation-oriented Simulation 78

4 Data Reconciliation Development **81**

4.1 Introduction 81
4.1.1 Objectives 81
4.1.2 Data Reconciliation Description 81
4.2 Sequential Modular Simulation 84
4.2.1 Optimization Algorithm 84
4.2.2 Objective Function 85

4.2.3 Manipulated Variable Selection 87
4.3 Equation-oriented Simulation 87
4.3.1 Optimization Algorithm 89
4.3.2 Objective Function 89
4.3.3 Manipulated Variable Selection 91
4.4 Discussion 93
4.4.1 Sequential Modular Simulation 93
4.4.2 Equation-oriented Simulation 96
4.4.3 Checking for Model Differences 99

5 Economic Optimization Development 101
5.1 Introduction 101
5.1.1 Objectives 101
5.1.2 Economic Objective Function 101
5.1.3 Process Inequality Constraints 104
5.1.4 Set Point Movement Restriction 105
5.2 Sequential Modular Simulation 105
5.2.1 Objective Function 108
5.2.2 Process Inequality Constraints 109
5.2.3 Manipulated Variable Selection and Bounding 109
5.3 Equation-oriented Simulation 111
5.3.1 Optimization Algorithm 111
5.3.2 Objective Function 112
5.3.3 Process Inequality Constraints 113
5.3.4 Manipulated Variable Selection and Bounding 113
5.4 Discussion 116
5.4.1 Sequential Modular Simulation 118
5.4.2 Equation-oriented Simulation 119
5.4.3 Checking for Model Differences 121

6 Real Time Optimizer Linking and Results 123
6.1 Real Time Optimizer Linking 123

6.1.1 Sequential Modular Simulator . . . 126
6.1.2 Equation-oriented Simulator . . . 127
6.2 Real Time Optimizer Comparison . . . 127
6.3 Economic Model Changes . . . 134
6.3.1 Product Price Changes . . . 135
6.3.2 Economic Objective Function Sensitivity Analysis . . . 137
6.3.3 Changes in Economics During Real Time Optimization Runs . . . 140
6.4 Process Disturbances . . . 146
6.5 ASPEN PLUS Software Upgrade . . . 148
6.6 Discussion . . . 152
6.6.1 Real Time Optimizer Linking . . . 152
6.6.2 Real Time Optimizer Comparison . . . 152
6.6.3 Economic Model Changes . . . 157
6.6.4 Process Disturbances . . . 160
6.6.5 ASPEN PLUS Software Upgrade . . . 161

7 Conclusions **163**
7.1 Real Time Optimizer Development . . . 163
7.2 Real Time Optimizer Comparison . . . 167
7.3 Process Specific Conclusions . . . 171
7.4 Recommendations for Future Research . . . 174

Bibliography **176**

A Real Time Optimizer Run Data **189**

B ASPEN PLUS Real Time Optimizer Documentation **237**
B.1 Purpose . . . 237
B.2 Specification . . . 237
B.3 Description . . . 239
B.4 References . . . 241
B.5 Files . . . 241

B.6 Input File Specifications . . . 243
B.6.1 IMEAS.DAT, IVARNC.DAT and IWGHT.DAT . . . 243
B.6.2 IDFREE.DAT . . . 243
B.6.3 ISMST.DAT . . . 246
B.6.4 IOBJFN.DAT . . . 246
B.6.5 ICNSTR.DAT . . . 246
B.6.6 IOFREE.DAT . . . 246
B.6.7 RVARNC.DAT . . . 248
B.7 Results . . . 248
B.8 Example . . . 252
B.8.1 Input Files . . . 252
B.8.2 Command Line . . . 262
B.8.3 Results . . . 262

C SPEEDUP Real Time Optimizer Documentation 267
C.1 Purpose . . . 267
C.2 Specification . . . 267
C.3 Description . . . 269
C.4 References . . . 271
C.5 Files . . . 272
C.6 Input File Specifications . . . 272
C.6.1 IMEAS.DAT, IVARNC.DAT and IWGHT.DAT . . . 272
C.6.2 IDFREE.DAT . . . 275
C.6.3 ISET.DAT . . . 275
C.6.4 IOBJFN.DAT . . . 278
C.6.5 ICNSTR.DAT . . . 278
C.6.6 IOFREE.DAT . . . 278
C.6.7 RVARNC.DAT . . . 282
C.7 Results . . . 282
C.8 Example . . . 283
C.8.1 Input Files . . . 283
C.8.2 Command Line . . . 289

C.8.3 Results 289

D ASPEN PLUS Real Time Optimizer Code **295**

D.1 setup 295

D.2 run 296

D.3 rto6.inp 300

E SPEEDUP Real Time Optimizer Code **355**

E.1 setup 355

E.2 run 357

E.3 Common Code from rtodr6.speedup and rtopt6.speedup . . . 362

E.4 Additional rtodr6.speedup Code 369

E.5 rtodr6.f 379

E.6 Additional rtopt6.speedup Code 385

E.7 rtopt6.f 398

E.8 model6x.speedup 407

E.9 report6y.speedup 422

E.10 prop6.inp 458

E.11 rtodr6setup.cmd 461

E.12 rtodr6run.cmd 461

E.13 rtopt6setup.cmd 462

E.14 rtopt6run.cmd 462

F Plant Simulation Code **465**

F.1 plant6.inp 465

F.2 randdata.f 491

List of Tables

1.1 Commercial real time optimization systems in development . . 7

3.1 Comparison of both process simulators for model development 76
3.2 Discrepancies between the ASPEN PLUS process model and Shell's RTO process model 79
3.3 Discrepancies between the SPEEDUP process model and Shell's RTO process model 80

4.1 ASPEN PLUS data reconciliation manipulated variables . . . 88
4.2 SPEEDUP data reconciliation manipulated variables 92
4.3 Comparison of both process simulators for data reconciliation 94
4.4 Discrepancies between ASPEN PLUS and SPEEDUP process models at the SPEEDUP data reconciliation optimum 100

5.1 Stream and utility unit prices 104
5.2 Economic optimization inequality constraints 106
5.3 ASPEN PLUS economic optimization manipulated variables . 110
5.4 SPEEDUP economic optimization manipulated variables . . . 114
5.5 Comparison of both process simulators for economic optimization 117
5.6 Discrepancies between ASPEN PLUS and SPEEDUP process models at the SPEEDUP economic optimization optimum . . 122

6.1 ASPEN PLUS economic objective function sensitivity study results 141

A.1 Base case ASPEN PLUS real time optimizer run 190
A.2 Base case SPEEDUP real time optimizer run 192
A.3 Base case with no measurement noise ASPEN PLUS real time optimizer run 194
A.4 Base case with no measurement noise SPEEDUP real time optimizer run 200
A.5 ASPEN PLUS real time optimizer run with Heavy Iso price at $173.89/m^3 204
A.6 ASPEN PLUS real time optimizer run with Light Distillate price at $200.52/m^3 206
A.7 ASPEN PLUS real time optimizer run with Heavy Iso price at $173.89/m^3 and Light Distillate price at $200.52/m^3 208
A.8 Winter ASPEN PLUS v9.3 real time optimizer run 210
A.9 Winter with no measurement noise ASPEN PLUS v9.3 real time optimizer run 212
A.10 Winter to summer ASPEN PLUS v9.3 real time optimizer run 218
A.11 Winter to summer with no measurement noise ASPEN PLUS v9.3 real time optimizer run 222
A.12 ASPEN PLUS real time optimizer run with external condendser disturbances 228
A.13 ASPEN PLUS real time optimizer run with external condendser disturbances and no measurement noise 230
A.14 Base Case ASPEN PLUS v9.3 real time optimizer run 234

B.1 ASEPN PLUS real time optimizer files required 242
B.2 Variable order used in the ASPEN PLUS real time optimizer input files `IMEAS.DAT`, `IVARNC.DAT` and `IWGHT.DAT` 244
B.3 Data reconciliation free variable ranges used in the ASPEN PLUS real time optimizer input file `IDFREE.DAT` 245
B.4 Simulation settings used in the ASPEN PLUS real time optimizer input file `ISMST.DAT` 247

B.5 Optimization parameters used in the ASPEN PLUS real time optimizer input file IOBJFN.DAT 249
B.6 Constraint bounds used in the ASPEN PLUS real time optimizer input file ICNSTR.DAT 249
B.7 Optimization free variable ranges used in the ASPEN PLUS real time optimizer input file IOFREE.DAT 251
B.8 ASEPN PLUS real time optimizer results files 251

C.1 SPEEDUP real time optimizer files required 273
C.2 Variable order used in the SPEEDUP real time optimizer input files IMEAS.DAT, IVARNC.DAT and IWGHT.DAT 276
C.3 Data reconciliation free variable ranges used in the SPEEDUP real time optimizer input file IDFREE.DAT 277
C.4 Simulation settings used in the SPEEDUP real time optimizer input file ISET.DAT 279
C.5 Optimization parameters used in the SPEEDUP real time optimizer input file IOBJFN.DAT 280
C.6 Constraint bounds used in the SPEEDUP real time optimizer input file ICNSTR.DAT 280
C.7 Optimization free variable ranges used in the SPEEDUP real time optimizer input file IOFREE.DAT 282
C.8 SPEEDUP real time optimizer results files 283

NOTE TO USERS

Page(s) missing in number only; text follows. Microfilmed as received.

xx

UMI

List of Figures

1.1 Hierarchy of process control applications 4
1.2 Structure of a sequential modular process simulator 9
1.3 Structure of an equation-oriented process simulator 11
1.4 Real time optimizer process flow diagram 15

2.1 General structure for model based real time optimization . . . 18
2.2 General steady state detection program flowchart 21
2.3 Examination of residuals test 23
2.4 Error probability distribution for process measurements 24
2.5 General steps for processing measurement data 25
2.6 Methodology for selecting a robust parameter estimation scheme 35
2.7 Two phase approach to on-line optimization 47
2.8 Integrated plant/control/simulator system 49
2.9 Centralized vs. distributed optimization 51
2.10 Overall process control systems structure 52
2.11 Interaction of optimizing program in the plant system 58

3.1 Real time optimizer process flow diagram with process model highlighted 60
3.2 Stabilizer-splitter process flow diagram 61
3.3 ASPEN PLUS stabilizer-splitter process model 65
3.4 SPEEDUP stabilizer-splitter process model 68

4.1 Real time optimizer process flow diagram with data reconciliation highlighted 82
4.2 Stabilizer-splitter process flowsheet including measurement variable tag numbers 83
4.3 Flow diagram summarising a general SQP optimization algorithm 86
4.4 Flow diagram summarising the SRQP optimization algorithm 90

5.1 Real time optimizer process flow diagram with the economic optimization highlighted 102
5.2 Example optimization illustrating the use of manipulated variable bounds 107

6.1 Real time optimizer process flow diagram with the real time optimizer highlighted 124
6.2 Real time optimizer data transfer 125
6.3 Graph comparing base case with no measurement noise profits for both real time optimizers 129
6.4 Graphs of the set point changes made during the base case runs with no measurement noise (1 of 2) 131
6.5 Graphs of the set point changes made during the base case runs with no measurement noise (2 of 2) 132
6.6 Graph comparing base case profits with measurement noise for both real time optimizers 133
6.7 Graph comparing profits for different Heavy Iso and Light Distillate prices using ASPEN PLUS 136
6.8 Graphs of the set point changes made during the runs with different Heavy Iso and Light Distillate prices using ASPEN PLUS (1 of 2) 138
6.9 Graphs of the set point changes made during the runs with different Heavy Iso and Light Distillate prices using ASPEN PLUS (2 of 2) 139

6.10 Graphs comparing profits for winter case and winter to summer price change with and without noise 143
6.11 Graphs of the set point changes made during the winter case and winter to summer price change with no measurement noise runs (1 of 2) 144
6.12 Graphs of the set point changes made during the winter case and winter to summer price change with no measurement noise runs (2 of 2) 145
6.13 Graphs comparing profits for base case and condenser disturbance runs with and without noise 147
6.14 Graphs of the set point changes made during the base case and condenser disturbance with no measurement noise runs (1 of 2) 149
6.15 Graphs of the set point changes made during the base case and condenser disturbance with no measurement noise runs (2 of 2) 150
6.16 Graph comparing base case profits with measurement noise for ASPEN PLUS versions 9.2 and 9.3 151

B.1 Stabilizer-splitter process flowsheet 238
B.2 ASPEN PLUS real time optimization program flow chart . . . 240

C.1 Stabilizer-splitter process flowsheet 268
C.2 SPEEDUP real time optimization program flow chart 270

Chapter 1

Introduction

1.1 Objectives

The main objective of this research is to compare the use of closed and open form equation based process simulation packages as the plant models in real time optimization. By developing real time optimization systems in both closed (sequential modular) and open (equation-oriented) form equation process simulators, this research will also help to determine the feasibility of using an "off the shelf" process simulation package as the process model in real time optimization. Development of the two real time optimization systems will provide "test beds" for further research on real time optimization applications and performance. This research will also help to close the gap between industry and research in this field as there is very little literature available on real time optimization.

Two other areas of real time optimization will also be studied in this research:

1. Determine the effect of changes in the financial model on the real time optimizer;

2. Demonstrate the effects of process disturbances on the operation of the real time optimizer.

These two areas will help to illustrate the benefits of performing real time optimization on a chemical process.

1.2 Real Time Optimization

The terms real time optimization (RTO) and on-line optimization are used to indicate the continuous reevaluation and alteration of operating conditions of a process so that the economic productivity of the process is maximized subject to operational constraints [56]. Optimization of process operations is of considerable interest in industry due to increasing global competition and tightening product requirements [32, 34]. Real time optimization is an appealing concept because it is at the level of the control hierarchy at which business decisions are integrated into the operation (see Section 1.2.1 for more detail). Advances in the speed and power of computers at lower costs are making on-line optimization a more cost effective method of improving plant performance.

Latour [62] lists the benefits from steady-state real time optimization as:

- Improved product yields and/or quality;
- Reduced energy consumption and operating costs;
- Increased capacity of equipment, stream factors;
- Less maintenance cost and better maintenance of instrumentation;
- More efficient engineers and operators as a tool for process trouble shooting and operation;
- Tighter, lower cost process designs if new plant designs include a real time optimizer.

However, not all processes will benefit from real time optimization. Processes that experience a wide range of operating conditions due to variability in

product prices and material costs, large variations in ambient conditions, or economic trade-offs and constraints will benefit the most from real time optimization. Some examples of suitable processes are steam utility plants, fluidized catalytic cracking units and downstream gas separation units [87].

A real time optimizer will perform the following general steps:

1. Detect when the plant is sufficiently close to steady state to perform optimization;
2. Perform data rectification on the process data (data reconciliation and possibly gross error detection);
3. Read the data into the process model and optimize the set points for the plant using an economic objective function and operational constraints;
4. Download the new set points into the plant's process control system and return to step 1.

Details on how each of these steps can be performed and the methodologies being used at present are covered in Chapter 2.

1.2.1 Real Time Optimization and Control Hierarchy

Figure 1.1 graphically represents the hierarchy of process control applications. At the base of the pyramid is regulatory control. This is a basic requirement for all for all further applications higher up the pyramid as they all require the availability of current on-line data from the plant [52].

Advanced regulatory control is used mostly in situations where where the process is unstable and requires stability. Advanced regulatory control is mostly applied using the DCS (distributed control system) hardware or with a higher level device such as an application module.

Advanced control is the application of software and control algorithms to do some predictive analysis and local optimization of the process. Some model based multivariable control techniques include generic model control

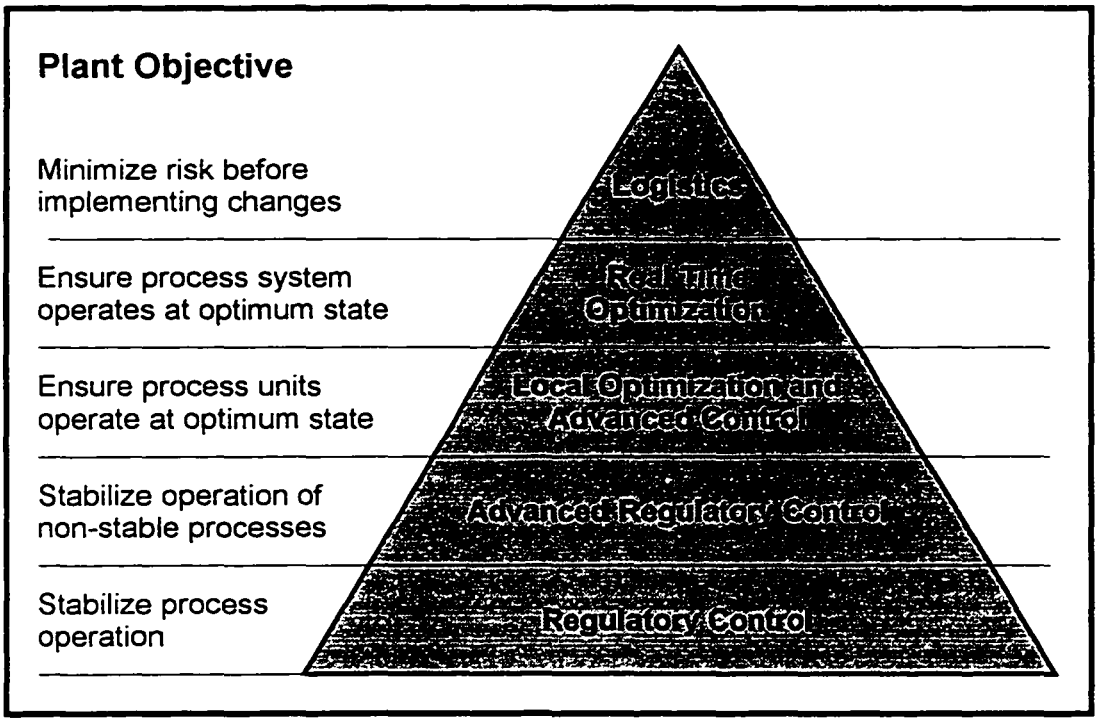


Figure 1.1: Hierarchy of process control applications [52].

(GMC) [22, 64], dynamic matrix control (DMC) [33], model algorithmic control (MAC) [90] and internal model control (IMC) [46]. Advanced controllers generally provide control strategies for one or two process units.

Wide area or real time optimization involves multiple process units, complex units or several areas within a plant. The goal is to maintain an optimum over a wider area of the plant and thus assumes that the combination of advanced and regulatory control can implement and maintain any set point requested by the optimizer.

Finally, at the top of the pyramid is "what-if" analysis. This includes production planning and scheduling as well as off-line analysis of the process.

1.2.2 Current Real Time Optimization Systems

Industrial real time optimization systems operating today are generally designed and coded specifically for each application. These large scale real time optimization systems are difficult to develop and maintain, requiring extensive development periods by highly skilled staff before implementation. The process models used in the real time optimizer are usually coded manually in open form equations, using a programming language such as FORTRAN 77, PASCAL or C. Typically, the process model will require in the order of 10,000 variables and equations to represent a chemical process. Subsequent maintenance of this process model is also difficult and highly specialized, often requiring recoding of sections of the process model. Historically, this method of development has been necessary to ensure that a cycle of the real time optimization can be completed within an acceptable amount of time on the computers available, typically between 1 and 3 hours.

Recently, computers have become powerful enough to allow the use of higher level programming and modelling techniques to be used in real time optimization. Several authors have proposed the use of process simulation packages for performing real time optimization [20, 45, 71, 105] (see Section 2.2.1 for further details). Both closed form (sequential modular) and open form (equation-oriented) equation process simulation packages have

been proposed. However, there are no known industrial applications of real time optimizers using process simulators. Historically, the reason for this was that the additional computational time required to complete real time optimization using a process simulation package was too great. With the recent advances in computer processor power and speed, this additional computational overhead is no longer a barrier to the use of process simulation packages in real time optimization systems. Process simulation packages also offer the advantages of built in unit operation models, powerful nonlinear optimization software and user friendly interfaces.

Several process simulation companies are working on the development of real time optimization and on-line modelling systems. These systems are summarised in Table 1.1. All of these systems are currently being implemented on chemical processes in conjunction with other companies, however it appears that these implementations are being used to assist in the development and testing of these packages. At the time of writing, none of these real time optimization systems had been released as commercial packages.

1.3 Process Simulation

Chemical process simulation, or process flowsheeting, is used to enable an engineer to model the behaviour of a plant, or parts of it, under defined conditions and so produce the heat and material balances for the process. In mathematical terms, the requirement is to solve the large number of nonlinear equations that represent the performance of the unit operations and the behaviour of the chemical components [104]. Process simulators are typically used in design, analysis and optimization of chemical processes [88].

Process flowsheeting architectures used to solve the nonlinear equations generated fall into two main categories: the sequential modular and the equation-oriented approaches. These two approaches and their advantages are discussed in the following subsections.

Table 1.1: Commercial real time optimization systems in development.

Company	**Product**	**Process Modelling Method**
Aspen Technology [14]	RT-OPT	Open form equation models, possibly based on SPEEDUP process simulation models.
Honeywell Hi-Spec Solutions [52, 53]	RT-EXEC (Previously OPTIMIZER)	Based on open form equation MASSBAL process simulation models.
HyproTech and MDC Technologies [55]	HYSYS.RTO+	Unknown.
Simulation Sciences [94]	ROM	Originally based on closed form equation PRO/II process simulation models, may now use open form equation models.
Treiber Controls	CRO	Open form equation models.

1.3.1 Sequential Modular Process Simulators

The sequential modular approach to process simulation implements unit operation blocks as computer subroutines, or modules, that calculate the output variables as functions of the input variables [67]. These computer subroutines are stored as library models which are matched to the various unit operations in the flowsheet. An executive program calls each subroutine in turn, using the output stream data from one unit as the input data for the next unit. Since most continuously operated plants involve multiple recycles, the executive program needs to manage the iterative calculation around the loops until a satisfactorily converged solution is achieved. Calls for physical properties are standardized so that consistent data are available throughout the program [104]. Thus, the basic components of a sequential modular simulation system are the executive, the model library and the physical properties system as summarised in Figure 1.2. The executive program accepts input data, determines the process flowsheet, derives and controls the calculation sequence for the unit operations in the flowsheet. The executive passes control to unit operations for the execution of the modules.

The level of model complexity is internalized within each model in a sequential modular simulation allowing the substitution of shortcut with rigorous unit models requiring only minor changes to the simulation program. Also, special purpose solution strategies can be constructed for each different unit type without altering the overall approach to the flowsheeting program. Liu et al. [67] list the advantages of sequential modular simulators as:

- the approach is conceptually easy for process engineers to understand;
- there is a large body of computer software organized in the form needed by the sequential modular approach;
- it is possible to include convergence heuristics developed by experience over the years (e.g., good initialization procedures);
- in case of an error, it easy to give understandable error messages.

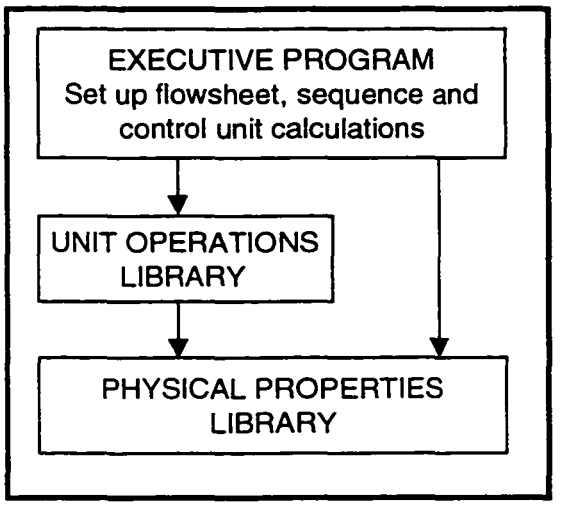


Figure 1.2: Structure of a sequential modular process simulator [18].

However, the input/output nature of the unit operation modules leads to difficulties as well. Problems occur when the recycle structure of the flowsheet and any external design specifications create awkward iteration loops, or tear streams, in the calculation sequence. Iteration loops are also required when any indirect unit specifications or design constraints are used [18].

Most of the more widely used flowsheeting systems such as Simulation Science's PRO/II, Aspen Technology's ASPEN PLUS and HyproTech's HYSIM employ the sequential modular approach (see Biegler [18] for a more comprehensive list of process simulation packages). The sequential modular approach has had wide acceptance due to the ease of building process flowsheets and the reliability of their solution.

Recently, some sequential modular process simulators have gradually added simultaneous modular methods. Unit operation modules remain intact, however the flowsheet stream connections are solved simultaneously using application of Broyden-type or quasi-Newton methods. This has allowed complicated recycles and flowsheet constraints to be solved together producing much faster convergence rates.

1.3.2 Equation-oriented Process Simulators

The basic idea of equation-oriented approaches to process simulation is to collect all of the equations describing the flowsheet together, and solve them as a large system of nonlinear algebraic equations. Mathematically this problem can be stated as [67]:

$$\begin{aligned} &\text{solve } f(\mathbf{x}, \mathbf{u}) = 0 \\ &\text{subject to } g(\mathbf{x}, \mathbf{u}) \leq 0 \end{aligned} \tag{1.1}$$

where $\mathbf{x}$ = the vector of state (dependent) variables;
$\mathbf{u}$ = the vector of decision (independent) variables;
$f(\mathbf{x}, \mathbf{u})$ = the set of process model equations;
$g(\mathbf{x}, \mathbf{u})$ = the set of inequality or equality constraints.

The decision variables include all of the input variables and the feed stream variables. The state variables include all the intermediate and product stream variables, internal variables within each unit operation block, and output performance variables from each block.

The crucial difference between sequential modular and equation-oriented process simulators is that the process equations, no matter how they are collected, are treated by general purpose solution strategies (e.g., Newton-Raphson) and are often solved simultaneously. The basic components of an equation-oriented process simulator are summarised in Figure 1.3. As seen in Figure 1.3, the executive performs a slightly different function to the executive for a sequential modular simulation (see Figure 1.2) in that it organises the equations and controls the general purpose equation solver.

Equation-oriented simulators are far more flexible than sequential modular simulators in terms of solving problems with nested tear recycle loops and additional design constraints at the flowsheet level. Also, the accessibility of derivative and function information within the equation oriented simulator allows for the application of more sophisticated optimization strategies. However, the solution of several thousand equations that represent a typical industrial scale process is a much more complex mathematical task than that

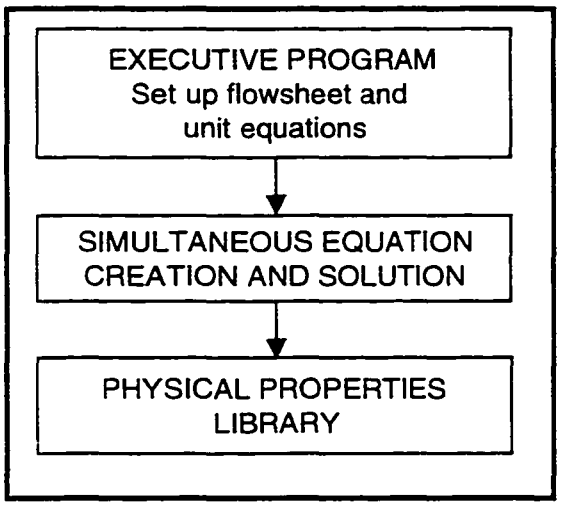


Figure 1.3: Structure of an equation-oriented process simulator [18].

posed by a sequential modular approach, where the equations are segregated into smaller sets in the unit operation routines. Equation-oriented flowsheeting problems require large scale numerical algorithms, good initialization strategies and reliable options to prevent convergence failures [18, 54, 104].

Equation-oriented process simulation packages are not as widely used as sequential modular simulators. Two commercially used packages are Aspen Technology's SPEEDUP, originally developed by Imperial College, and Honeywell Hi-Spec Solutions' MASSBAL, commonly used in the pulp and paper industry. Neither of these packages are "purely" equation-oriented as they incorporate procedures at the lowest levels to promote convergence. In particular, physical properties are determined using procedures rather than including large sets of nonlinear property equations in the simulation.

1.4 Scope of this Research

Current large scale real time optimizers implemented in industry usually suffer from a lack of user friendliness. Both Lojek and Whitehead [68], and

Darby and White [34] emphasize the importance of good man-machine interfaces required to facilitate the use and maintenance of a real time optimizer. One method of simplifying the development and subsequent maintenance of a real time optimization system is to use process simulation packages. Several process simulation packages already have the features required to implement a complete real time optimization system, including the ability to perform the process modelling, data reconciliation, economic optimization and data transfer required. In addition to simpler modelling environments, process simulation packages also offer the advantages of rigorous property calculations, choice of convergence algorithms and, with some packages, graphical user interfaces.

Process simulation packages also offer the advantage of providing a standardized modelling environment for several areas of the process control hierarchy shown in Figure 1.1. As well as providing the models for economic optimization and data reconciliation in real time optimization, the models could also potentially be used in advanced control and "what-if" analysis. For example, process simulator models can be used in Generic Model Control (GMC) as demonstrated by Douglas et al. [36] and Lee [65] who both used the SPEEDUP simulations package as the model. The same models used in the real time optimizer and advanced control can also be used by process engineers for general process simulation and trouble shooting. These same models may also be used for nonlinear modelling of entire plants for scheduling and production planning purposes. For example, McLellan [75] and Picaseno-Gamiz [86] both used SPEEDUP to present systematic nonlinear approaches to scheduling for continuous processes.

The purpose of this research is to compare the use of the two main types of process simulation package in real time optimization applications. The two main types are closed form equation, or sequential modular, and open form equation, or equation-oriented, process simulators. Both types of process simulator were used to develop a real time optimizer for the same chemical process to compare the development and operation advantages of each

package.

The two process simulation packages chosen were ASPEN PLUS v9.2 [8, 9] and SPEEDUP v5.5d [11, 12]. A newer version of ASPEN PLUS, version 9.3 was used for some real time optimizer testing, however, unless otherwise stated in the thesis, version 9.2 of ASPEN PLUS was used. These packages were chosen as being representative of their respective types of process simulator and for their extensive optimization capabilities.

ASPEN PLUS is a sequential modular process simulator with some simultaneous modular convergence options. It also has a wide range of unit operation models with tailored convergence methods for best performance. An SQP (successive quadratic programming) algorithm is available for optimization problems (see Section 4.2.1 for more detail on SQP) and user defined FORTRAN blocks can be added to a process flowsheet. The process flowsheet is specified either using the MODELMANAGER graphical user interface or by editing a keyword based text file.

SPEEDUP is an equation-oriented process simulation package with procedure base property calculations. SPEEDUP contains a library of precoded simple unit operation models which can be combined to form more complex units. Both MINOS and SRQP (successive reduced quadratic programming) optimization algorithms are included with SPEEDUP (see Section 4.3.1 for more detail on SRQP). User defined models may be added using both SPEEDUP's modelling code and additional FORTRAN procedures if required. The process flowsheet is specified by editing a keyword based text file, with a text based user interface providing syntax checking, result display and several error diagnosis tools.

The chemical process chosen for testing both real time optimizers was a stabilizer-splitter process at Shell Canada's Sarnia refinery. This process is used to remove the light ends from a hydrocracked stream in the stabilizer column, and then remove the petroleum products in the splitter fractionation column. For a detailed description of this process, see Section 3.1.2. The stabilizer-splitter was chosen as being a suitably complex industrial process

for testing a real time optimizer, while being small enough to be optimized within a reasonable period of time using the computing resources available.

The basic structure of the real time optimizer to be developed using both types of process simulator is summarised in Figure 1.4. Both real time optimizers were developed in four stages:

1. The process model to be used in both the data reconciliation and economic optimization was simulated. The development of the process model is discussed in Chapter 3.

2. The data reconciliation optimization was added to fit the measured data to the process model using a least squares objective function optimization. Gross error detection was not performed on the process data. See Chapter 4 for details of the data reconciliation development.

3. The economic optimization using a profit objective function was developed. This optimization development is discussed in Chapter 5.

4. Finally, the above components were linked together to form the real time optimizer as shown above the dashed line in Figure 1.4. The plant, shown below the dashed line in Figure 1.4, provides process measurements either from real process data supplied by Shell, or from a steady state process simulation with noise added to the results. See Section 6.1 for details on the formation of the real time optimizers.

The testing of and results obtained from the two real time optimizers are also discussed in Chapter 6.

The development of the two real time optimization applications will also provide "test beds" for future on-line optimization testing and research. To date, most research into real time optimization has been conducted in industry, and there has been very little academic research into this area. The availability of a working real time optimization application should help close the gap between industry and research in the field of real time optimization.

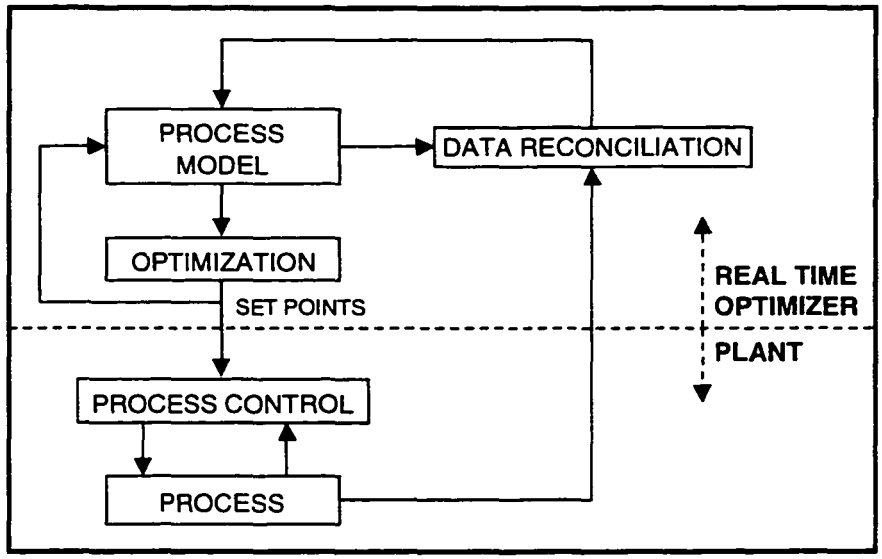


Figure 1.4: Real time optimizer process flow diagram.

In addition to comparing the two types of process simulators in real time optimization applications, two further areas of real time optimization were studied. The first is to study the effects of changes in the economic objective function on the real time optimizer. The results of this study is presented in Section 6.3. The second area is to demonstrate the effect of process disturbances on the operation of the real time optimizer (see Section 6.4 for details). The study of these areas will help to illustrate the benefits of performing real time optimization on this process when changes in either economic or process conditions occur.

1.5 Conventions Used in this Thesis

Real time optimizers and process simulators perform several different types of iterations during their operation. To prevent confusion about which type of iteration is being discussed, the following conventions will be used:

- "Cycle" will be used to indicate a full cycle of the real time optimizer. This is one complete run through the flow chart given in Figure 1.4, that is the measured data is read from the plant, data reconciliation is performed on this data, then the process is optimized before the set points are passed back to the plant.

- "Iteration" will normally refer to an iteration of an optimization algorithm (either SQP in ASPEN PLUS or SRQP in SPEEDUP). Iterations occur in both the data reconciliation optimization and the economic optimization steps of a real time optimization cycle.

- Flowsheet "passes" refer to one evaluation of the entire process flowsheet by the simulator. Multiple flowsheet passes are usually made on an iteration of an optimization algorithm.

- "Tear stream iterations" are required in ASPEN PLUS to converge a tear stream. To prevent confusion with optimization algorithm iterations, these will always be referred to in full as tear stream iterations. Several tear stream iterations are required on each flowsheet pass to converge a tear stream.

Chapter 2

Literature Review

2.1 Components of a Real Time Optimizer

A general structure for real time optimization based on the one shown in Hanmandlu et al. [49] is given in Figure 2.1. This structure is also similar to those presented in several other papers [17, 34, 38, 43, 44, 56, 68, 70, 83, 93].

The general steps taken in one cycle of the real time optimizer pictured in Figure 2.1 are:

1. *Steady State Detection:* This detects when the process is close enough to steady state to allow the real time optimizer to model the process with reasonable accuracy. When the plant is determined to be at steady state, data from the process is passed onto the optimizer.

2. *Data Reconciliation and Gross Error Detection:* All measured data are subject to error. Data reconciliation adjusts the measured variables (e.g., energy, mass, etc.) and, if possible, estimates any unmeasured variables for the process so that they satisfy the balance constraints. Gross error detection takes suitable corrective actions to rectify any gross errors found in the process data [101]. This adjusted data from the plant is then passed onto the next step.

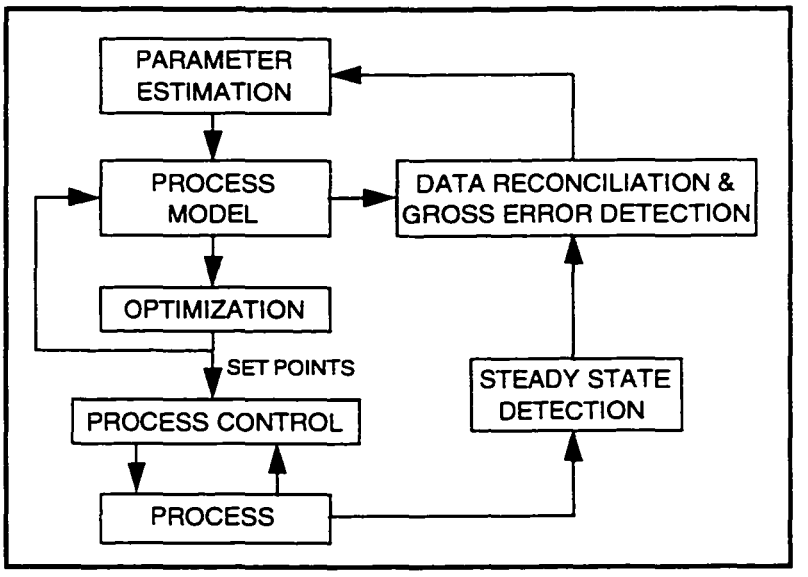


Figure 2.1: General structure for model based real time optimization.

3. *Parameter Estimation:* The process of updating the model parameters to the adjusted process data available. These model parameters are then used in the process model for the optimization of the set points.

4. *Process Model:* The model of the process which represents the effects of changes made to the process by the optimization algorithm while it is iterating to an optimum solution.

5. *Optimization:* The optimization algorithm used to determine the optimum set points for the process given an objective function and process constraints.

6. *Updating of Process Set Points:* The optimum set points for the process found by the optimizer are returned to the plant. Once the plant has returned to steady state, another cycle of the real time optimizer is performed to keep the plant operating at its optimum.

Some of the above steps are detailed further in the following subsections.

2.1.1 Steady State Detection

Steady state is one of the most important assumptions made when designing a real time optimizer because the plant is being represented by a steady state model [82]. The steady state model is used to perform data reconciliation and parameter estimation on the raw data from the plant. If unsteady or transient data is used, the reconciled results may not accurately reflect the operation of the plant, and therefore the subsequent optimization of the plant may not yield a realistic optimum for the current operating conditions. For this reason, a method of checking for steady state is required before real time optimization can proceed.

In any steady state detection algorithm, two questions must be answered [87]:

1. Which measured variables should be used to determine if the process is at steady state? Not every measured variable should be analysed in assessing the state of the plant. The variables used to assess the plant condition should be chosen from the measured variables considered to be the most representative of the state of the plant. These choices are usually made based on operational experience within the plant.

2. How much variability in the measurements is allowed before the process is deemed unsteady? This question is also answered from plant operating experience. When the plant is considered to be at steady state over a period of time, the variances of the key variables can be estimated. These base variances can then be used to determine the plant's "steadiness" when compared to the future observed variances. Generally, on-line tuning of the steady state detection algorithm will be required over a period of time to ensure that the state of the plant is being predicted accurately.

Two methods for determining the condition of the plant have been proposed in the literature; the successive means test and examination of residuals. The successive means test offers a more statistically valid method of checking for steady state, while the examination of residuals method has a

higher degree of tunability. Figure 2.2 shows a general steady state detection program algorithm.

Successive Means Test:

Two papers by Narasimhan et al. [80. 82] detail the successive means test method. This test involves the calculating the means and variances of the key variables over two time periods. These two periods are then compared statistically to determine if there is a significant difference between the two means. This statistical test is based on the mathematical theory of evidence and is calculated using the following calculation [80]:

$$t_{1,i}^2 = N\left(\frac{(\bar{x}_{2,i} - \bar{x}_{1,i})^2}{s_{1,i}^2 + s_{2,i}^2}\right) \tag{2.1}$$

where $t_{1,i}^2$ = a random variables obeying Hoteller's T^2 distribution with numerator degrees of freedom 1. and denominator degrees of freedom $2N - 2$.

N = number of measurements in each period.

$\bar{x}_{1,i}$ = mean value of variable i in the first time period.

$\bar{x}_{2,i}$ = mean value of variable i in the second time period.

$s_{1,i}^2$ = sample variance of variable i in the first time period.

$s_{2,i}^2$ = sample variance of variable i in the second time period.

A level of significance, α, is chosen, and $T^2(\alpha)$ is calculated to be the upper quantile of the T^2 distribution. Two cases are now possible:

Case I: $t_{1,i}^2 \leq T^2(\alpha)$, and therefore the plant is presumed to be at steady state with respect to variable i.

Case II: $t_{1,i}^2 > T^2(\alpha)$, and therefore the plant is presumed to not be at steady state.

The following assumptions are made in using this method:

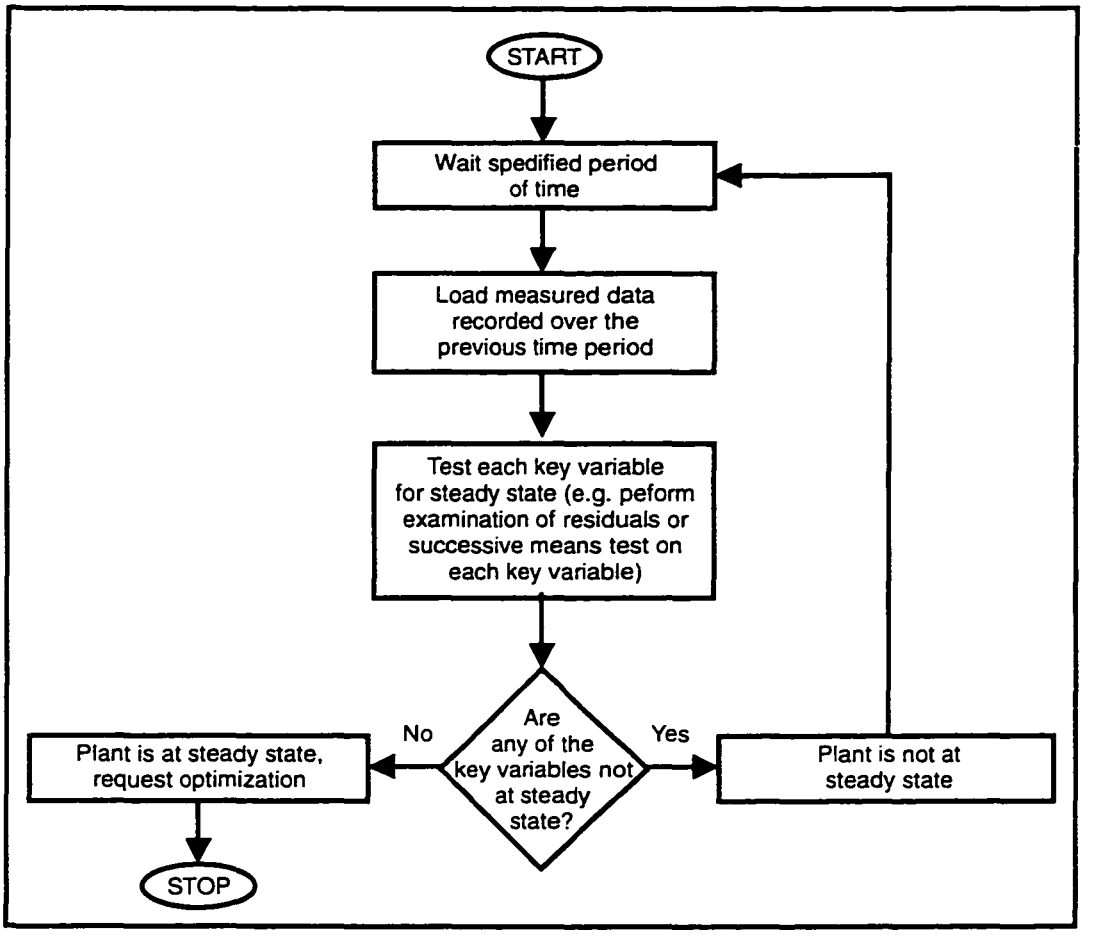


Figure 2.2: General steady state detection program flowchart [87].

1. A process undergoes a change in steady state if the true values of one or more of its variables change.

2. Measurements of process variables only contain normally distributed random errors with mean $\underline{0}$ and covariance matrix $\mathbf{Q}$.

3. A time period consists of N successive measurements. The process state can change from one period to another, but within a period the process is assumed to be in a steady state.

4. Successive measurements are mutually independent.

5. $\mathbf{Q}$ is unknown. It is diagonal and constant from one period to another.

Examination of Residuals:

Examination of residuals is based on a confidence region approach to determine if the process is at steady state. A maximum residual value is chosen for each variable. The number of violations of this maximum residual are determined from the sample of N measurements over a given period. If the number of violations for a variable exceeds the maximum violations allowed, then the plant is assumed to not be at steady state. If the maximum violations allowed is not exceeded then the plant is presumed to at steady state with respect to that variable. Figure 2.3 illustrates how the examination of residuals method determines both steady and non-steady state variables.

2.1.2 Data Reconciliation and Gross Error Detection

The key features of the process data problem are as follows:

1. All measurements are subject to errors. Random errors are assumed to be normally distributed and have zero mean. Gross errors are caused by non-random events such as instrument biases, malfunctioning measuring devices, inaccurate or incomplete process models and process

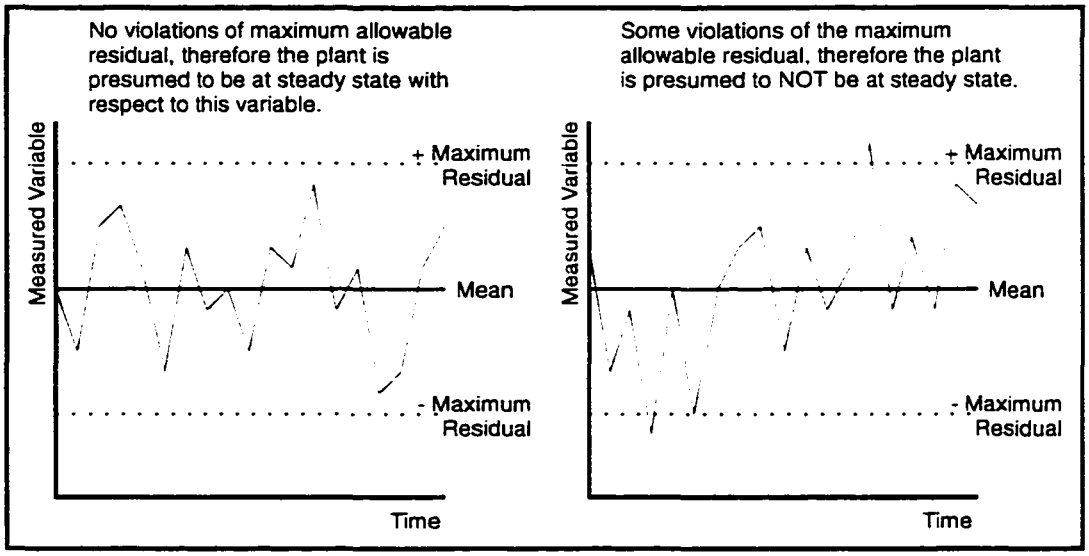


Figure 2.3: Examination of residuals test [52].

leaks [66, 73, 101]. Figure 2.4 gives an illustration of the probabilities involved for random and gross errors.

2. Not all process variables are measured for reasons of cost, inconvenience or technical infeasibility.

3. There is data redundancy in the sense that there are more measurements (or data) available than needed if the measurements were not subject to errors. This is also referred to as spatial redundancy [28, 73, 76].

4. Since most process data are being sampled continuously and regularly at great frequencies, there is also data redundancy if the process conditions are truly at steady state. This is usually referred to as temporal redundancy.

The adjustment of measured variables for random errors and, if possible, the estimation of unmeasured variables so that they satisfy the balance

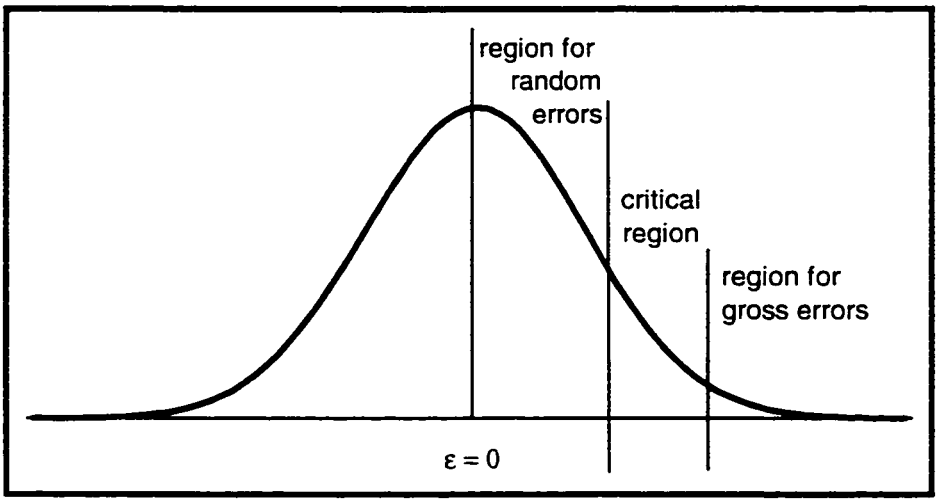


Figure 2.4: Error probability distribution for process measurements [101].

constraints is known as the data reconciliation problem. Detection of the presence of any gross errors so that suitable corrective actions can be taken is known as the gross error detection problem [99]. Collectively, these two problems are sometimes called data rectification [100]. The objectives of data rectification are to improve confidence in the measurements, estimate unmeasured streams, identify meter faults and to identify process losses [63]. A simplified view of the three basic steps to data rectification is shown in Figure 2.5. The first step, variable classification, involves determining which variables are observable and/or redundant [66]. In practice, steps two and three are often used iteratively to reconcile the data and remove gross errors.

Linear, Steady State Data Reconciliation:

The simplest case is linear, steady state data reconciliation. This is covered by Mah [73] and Crowe et al. [31]. The basic model used for the data reconciliation in the absence of gross errors [73] is:

$$\mathbf{y} = \mathbf{x} + \varepsilon \tag{2.2}$$

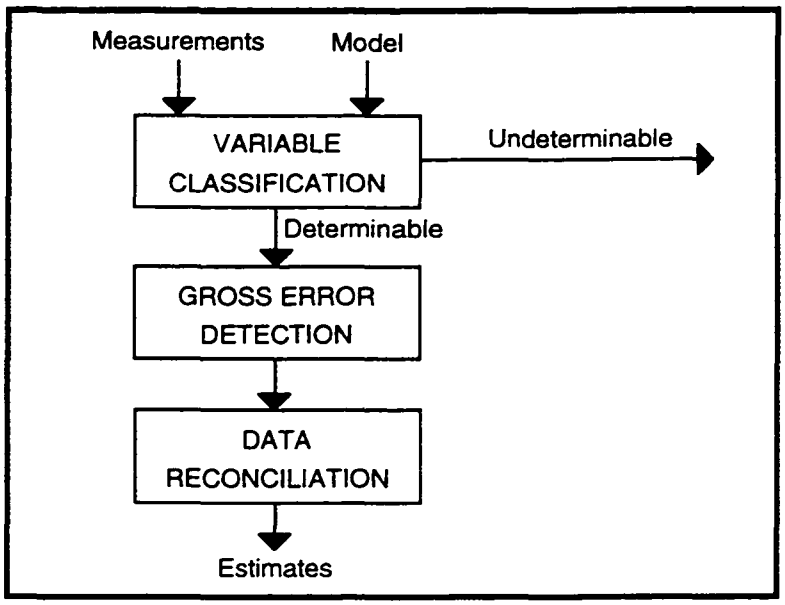


Figure 2.5: General steps for processing measurement data [66].

where $\mathbf{y}$ = $(s \times 1)$ vector of measured variables;
$\mathbf{x}$ = $(s \times 1)$ vector of true variable values;
ε = $(s \times 1)$ vector of random measurement errors.

It is usually assumed that:

1. the expected value of ε, $\mathrm{E}(\varepsilon) = 0$;
2. the successive vectors of measurements are independent, i.e., $\mathrm{E}(\varepsilon_i \varepsilon_j^{\mathrm{T}}) = 0$, for $i \neq j$;
3. the covariance matrix is known and positive definite, i.e., $\mathrm{cov}(\varepsilon) = \mathrm{E}(\varepsilon_i \varepsilon_i^{\mathrm{T}}) = \mathbf{Q}$, and $\mathbf{Q}$ is positive definite and known.

The linear constraints are formed by stoichiometric constraint, energy, mass and other balances. In this case they are linear and homogeneous, and given by:

$$\mathbf{Ax} = \mathbf{0} \tag{2.3}$$

where $\mathbf{A}$ = $(n \times s)$ coefficient matrix representing the linear process model ($\mathbf{A}$ is of full row rank).

If we assume that the measurement errors are normally distributed, the data reconciliation problem may be formulated as the following constrained least squares estimation problem:

$$\begin{aligned} &\min_{\mathbf{x}} \left[(\mathbf{y}-\mathbf{x})^{\mathrm{T}}\mathbf{Q}^{-1}(\mathbf{y}-\mathbf{x})\right] \\ &\text{subject to } \mathbf{Ax}=\mathbf{0} \end{aligned} \tag{2.4}$$

The solution is given by:

$$\hat{\mathbf{x}} = \mathbf{y} - \mathbf{QA}^{\mathrm{T}}\left(\mathbf{AQA}^{\mathrm{T}}\right)^{-1}\mathbf{Ay} \tag{2.5}$$

Crowe et al. [31] use a projection matrix to obtain a reduced set of balance equations (equation (2.3)), and also include a statistical test for the removal of gross errors. Mah [73] also covers the decomposition and solution of these equations in detail.

Nonlinear, Steady State Data Reconciliation:

Nonlinear, steady state data reconciliation is the extension of the above method to include nonlinear constraints. These occur when variables are measured indirectly by other physical properties, e.g., concentration by density, pH or thermal conductivity. Bilinear constraints occur when variables appear in two balances, e.g., temperatures measured along with concentrations will appear in both energy and component material balances. Bilinear constraints also occur when a stream is split into two or more streams of the same temperature and composition. No general analytical solution is available to bilinear and nonlinear data reconciliation problems. Nonlinear constraints also make it more difficult to determine whether unmeasured variables may be estimated, and how to decompose a problem with missing measurements [73].

One simple method of solving the nonlinear data reconciliation problem is to extend the linear case to handle the nonlinear constraints [87]:

$$\begin{aligned} &\min_{\mathbf{x}}[(\mathbf{y}-\mathbf{x})^{\mathrm{T}}\mathbf{Q}^{-1}(\mathbf{y}-\mathbf{x})] \\ &\text{subject to } h(\mathbf{x})=\mathbf{0} \\ &\qquad\qquad g(\mathbf{x})\geq\mathbf{0} \end{aligned} \tag{2.6}$$

where $h(\mathbf{x})$ = the set of equality constraints representing the process model:
$g(\mathbf{x})$ = any inequality constraints present in the process model.

There is no longer an analytical solution to the problem given in equation (2.6) and the problem must be solved using an optimization algorithm. In practical applications of this method, weights are added to the least squares objective function to place emphasis on key measured variables. A large weighting factor for a given variable will force the optimization to reduce the degree of adjustment for that variable. These weights are usually chosen based on operational experience. The objective function for the nonlinear data reconciliation then becomes:

$$\begin{aligned} &\min_{\mathbf{x}}\left[\sum_{i=1}^{n} w_i\left(\frac{y_i-x_i}{s_i}\right)^2\right] \\ &\text{subject to } h(\mathbf{x})=\mathbf{0} \\ &\qquad\qquad g(\mathbf{x})\geq\mathbf{0} \end{aligned} \tag{2.7}$$

where w = weighting factor associated with each measured variable;
y = the measured variable;
x = the true variable value;
s = the standard deviation of the measurements (usually the standard deviation of the measurement transmitter).

Crowe [26] handles the nonlinear constraints by extending the method outlined in Crowe et al. [31] by applying two successive projection matrices to the equations defining conservation laws and other constraints. Also, no restrictions are placed on the location of measurements, nor upon what is

measured in any one stream. The construction of the projection matrices is direct and each requires the inversion of a matrix that is also needed for subsequent steps in the solution.

A paper by Serth et al. [89] adds details on the choice of regression variables used in nonlinear data reconciliation by defining primary and secondary variables. Total flow and component flows are examples of primary and secondary flows respectively. Since material balances are linear in secondary variables, they can be exploited to develop a particularly efficient data reconciliation procedure [26]. However, they do not yield maximum likelihood estimates for the true values of primary measured variables. This in turn could affect the error detection procedure, tending to increase the number of Type I errors (Definition of Type I error is given in the Gross Error Detection section).

Data reconciliation using process simulation software has been looked at by both Macchietto et al. [70] and Stephenson and Shewchuk [95]. Both used equation based simulation packages. Macchietto et al. [70] used both SPEEDUP and a package using simplified process models called DEBIL. SPEEDUP was found to be slower and use more computational resources, but was more flexible. Stephenson and Shewchuk [95] detail the methodology by which MASSBAL solves the data reconciliation problem. Applications to a pulp mill screening and cleaning system are described.

Takiyama et al. [98] outline the use of SEBDARM (Sensor Based Data Reconciliation Method). The method is based on the direct use of measurement variables rather than using balanced variables (conventional method). This has the advantage of having the reconciled data not being influenced by the number of reduced balance equations. Their method, however, has only been tested on pilot plants to date.

Dynamic Data Reconciliation

This is relatively new area of research in the field of data reconciliation. In many practical situations the process conditions are continuously undergoing

changes and steady state is never truly reached. Generally the data reconciliation problem in this case is reduced to a discrete Kalman filter for a quasisteady-state problem [35].

Darouach and Zasadzinski [35] develop a new dynamic algorithm based on steady state reconciliation leading to a recursive scheme useful for real time processing of data. The method was shown to reduce computational problems associated with discrete Kalman filters, especially for singularities and round off error situations. Rollins and Devanatham [89] refine the method proposed by Darouach and Zasadzinski [35] to reduce the computational time required, while preserving its properties of unbiased estimation. Liebman et al. [66] use nonlinear programming to help solve systems in highly nonlinear regions where the Kalman filter method may be inaccurate. The method given was found to be a more robust means for reconciling dynamic data, especially in the presence of inequality constraints. The method is demonstrated on a reactor example.

Gross Error Detection

Gross errors in the data invalidates the statistical basis of reconciliation due to their non-normality. One gross error present in a constrained least squares reconciliation will cause a series of small adjustments to other measured variables. For these reasons, gross errors need to be identified and removed before data reconciliation is carried out. The most common techniques for detecting gross errors are based on statistical hypothesis testing of the observed or measured data. For any gross error detection method to work, there must be at least two alternative ways of estimating the value of a variables, for instance, measured and reconciled values [73]. Crowe [27] and Mah [73] give detailed accounts of typical statistical gross error detection algorithms.

Two measures of the effectiveness of a gross error detection scheme are the terms Type I and Type II errors made by the detection scheme. Type I errors refer to wrongly identifying truly random errors to be gross errors. Type II errors refer to wrongly finding truly gross errors to be random errors [27].

In devising the tests for a gross error, the probability of a correct detection must be balanced against the probability of mispredictions.

The Maximum Power (MP) test for Gross Errors is detailed in two papers by Crowe [29, 30]. The maximum power test has the greatest probability of detecting the presence of a gross error without increasing the probability of Type I error than any other test would on a linear combination of the measurements. Crowe [29] only deals with the linear case, while Crowe [30] extends the method to the bilinear case and retests the original constraints used.

The Generalized Likelihood Ratio (GLR) approach is detailed by Narasimhan and Mah [81] and Mah [73]. The previously discussed methods only deal with gross errors from measurement or sensor biases. The GLR approach provides a framework for identifying any type of gross errors that can be mathematically modeled. This can be very useful for identifying other process losses such as a leak. The method and examples of its use are detailed reasonably fully by Narasimhan and Mah [81]. Narasimhan [78] shows that the Maximum Power Test detailed above is equivalent to the GLR approach for simple steady state models.

Serth and Heenan [91] tested 3 different methods of gross error detection on the same steam metering system. These methods were:

1. Measurement test (MT) method. This method proposed by Mah and Tamhane [74] uses a test which possesses maximal power properties for detecting the presence of a single outlier (gross error) of the least squares residuals. The main problem with this method is that the least squares procedure tends to spread the gross errors over all the measurements, thereby creating large residuals corresponding to good measurements [91]. Two modifications of the MT have also been proposed: the iterative measurement test (IMT) method, and the modified iterative measurement test (MIMT) method. The IMT method simply applies the MT method iteratively removing one gross error at each iteration until none remain. IMT, however, can predict sets of recon-

ciled variables which may contain negative flow rates or absurdly large values. The MIMT adds bounds to the reconciled variables to prevent negative flows and absurdly large values

2. Method of pseudonodes (MP). This method proposed by Mah et al. [72] is based on a nodal imbalance test applied to each node and also to aggregates of two or more nodes, which are called pseudonodes. This method predicts potentially bad streams (i.e. may contain gross errors) and then treats these streams as being bad. Therefore this method will tend to err on the side of treating good streams as containing gross errors [91]. A modified method of pseudonodes (MMP) fixes a problem which arises in MP when no potentially bad streams are predicted, but one or more nodes are bad. Practically, this modification has very little effect on the performance of this method.

3. Combinatorial method. The basic idea behind this method proposed by Serth and Heenan [91] is to identify combinations of gross errors that are consistent with the observed pattern of nodal imbalances. The Combinatorial method was found to be less susceptible to error cancellation than MP or MMP as the algorithm does not work with aggregates of nodes. However, this method can result in an extremely large combinatorial problem, even on on a system of moderate size. The Screened Combinatorial (SC) method was proposed to alleviate this problem by using the MMP method as a screening procedure to reduce the size of the combinatorial problem.

From the results of their testing, Serth and Heenan [91] concluded that both the screened combinatorial (SC) and modified iterative measurement test (MIMT) methods constitute effective and reliable methods for gross error detection and data reconciliation in steam metering systems. MIMT was found to be computationally more efficient than the SC method. Both the MT method and MP were found to perform very poorly.

Combined Data Reconciliation and Gross Error Detection Methods

Two papers by Narasimhan and Harikumar [50, 79] give details of incorporating bounds into data reconciliation. Most approaches to data reconciliation do not impose inequality constraints because the solution can no longer be obtained analytically, and the statistical analysis of the measurement residuals for gross error detection is difficult. This leads to an iterative cycle of data reconciliation and gross error detection phases to complete the data rectification. Narasimhan and Harikumar [79] develop the algorithm for solving the data reconciliation problem with bounds and produce the constraint and measurement residuals. In the second paper [50], the gross error detection procedures are detailed and the whole method is compared to existing methods. Their results indicate that their method performed better than currently available methods.

Tjoa and Biegler [101] develop a simultaneous data reconciliation and gross error detection method. They minimize an objective function that is constructed using maximum likelihood principles to construct a new distribution function. This distribution takes into account both contributions from random and gross errors. This method gives unbiased estimates in the presence of gross errors. Therefore simultaneously a gross error detection test can be constructed based on the new distribution functions without the assumption of linearity of constraints. This method is particularly effective for nonlinear problems.

Serth et al. [92] show how the choice of regression variables can adversely affect the performance of an associated gross error detection algorithm. The principal effect is an increased tendency of the algorithm to make Type I errors. Their study was carries out on two sets of variables: primary or total flow variables, and secondary or components flow rates. They conclude that for systems with relatively small numbers of variables and constraint equations, data reconciliation should be performed by regressing the primary variables. For large systems where computational efficiency has a high priority, the best approach may be to regress on the secondary variables and

to use the primary residuals for gross error identification.

Terry and Himmelblau [100] detail the use of Artificial Neural Networks (ANN) for Data reconciliation and gross error detection. The ANN is "trained" or "learns" to perform the data rectification based on a given model and constraints. The method is still being researched and the results look promising compared to models built from first principles.

2.1.3 Parameter Estimation

Parameter estimation, or model parameter updating, involves updating of any process parameter in the model which changes slowly over time, usually due to wear and tear of the equipment or catalysts. Process parameters include heat transfer coefficients, column efficiencies and loading factors, and reactor effectiveness factors. The real-time updating of a model's parameters permits realistic models to be provided for an optimization system. Another benefit of parameter estimation is the ability to track the condition of equipment in the plant with time (e.g., heat exchanger fouling, catalyst activity etc.) [32].

A simple method of updating process parameters is to include them in the objective function during data reconciliation. Since these parameters should not vary significantly between each cycle of a real time optimizer, movement from the previous parameter value should be penalized. This can be achieved by setting the weight associated with a parameter to a large value (e.g., two or three orders of magnitude larger than the weights associated with measured variables) [87]. In this case, the nonlinear constrained data reconciliation problem given in equation (2.7) becomes:

$$\begin{aligned} \min_{\mathbf{x},\mathbf{p}} & \left[\sum_{i=1}^{n} w_i \left(\frac{y_i - x_i}{s_i} \right)^2 + \sum_{j=1}^{m} v_j \left(p'_j - p_j \right)^2 \right] \\ \text{subject to } & h(\mathbf{x}) = \mathbf{0} \\ & g(\mathbf{x}) \geq \mathbf{0} \end{aligned} \tag{2.8}$$

where $v =$ weighting factor associated with each parameter to be estimated;

p' = the previous estimated parameter value;
p = the estimated parameter value.

MacDonald and Howat [71] propose two different methods of combined data reconciliation and parameter estimation. The first method is a sequential, "decoupled" procedure that reconciles the data to satisfy the material and energy balances, and then estimates the process parameters using maximum likelihood estimation. The second method is a "coupled" procedure that simultaneously reconciles the data to satisfy the constraints and estimates the process parameters. Both methods were tested on a single stage flash. It was found that the decoupled procedure was computationally faster and more easily adapted to existing reconciliation algorithms. However, the coupled procedure was more statistically rigorous and gave better parameter estimates.

Krishnan et al. [59] present a two step method for the design of a robust parameter estimation scheme that may be used to update the process model employed in on-line optimization. This method is summarized in Figure 2.6. The first step involves a determining the "key" model parameters (i.e. those having a significant effect on the calculated optimum). The second step involves finding the "best" set of measurements to estimate these parameters. This is achieved by using structural analysis, singular value analysis and calculation of parameter confidence regions. A case study using this method is detailed in a later paper [60].

Stewart et al. [96] provide an extensive review of Bayesian and likelihood approaches to parameter estimation from various types of multiresponse data with an unknown covariance matrix. They found that these methods are preferable to weighted least squares and to use of a specified covariance matrix, since they estimate the covariance and parameters optimally from the data provided. Several software programs were described for performing both least squares and multiresponse parameter estimation. Some multiresponse applications were also reviewed.

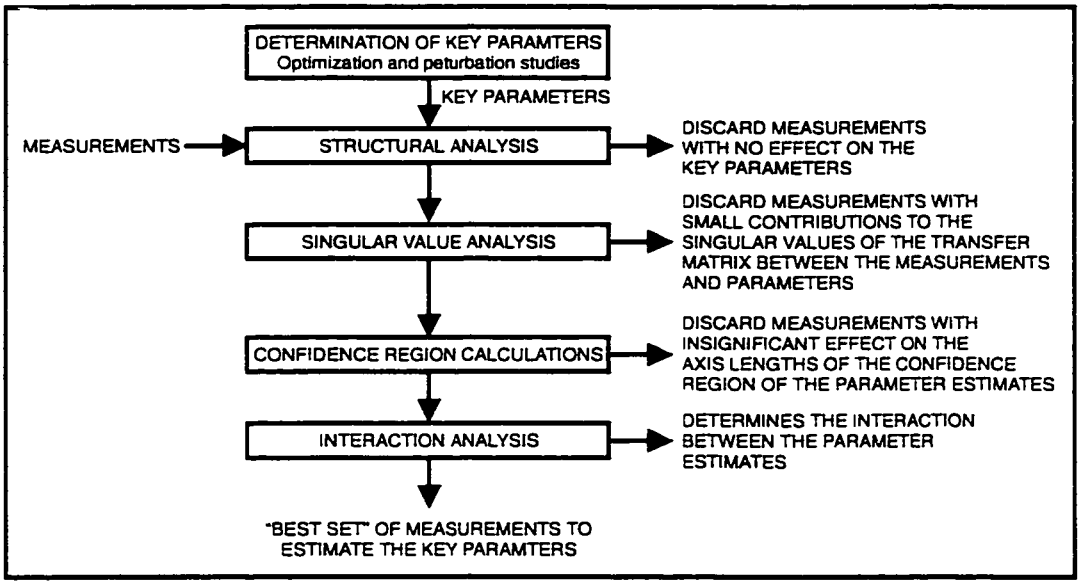


Figure 2.6: Methodology for selecting a robust parameter estimation scheme [59].

2.1.4 Process Model

The process model used in real time optimization must represent changes in the manipulated variables of the process given the updated set points calculated by the optimization algorithm. The model approximates the effect of these variables on the dependent variables, which are usually those that are constraints or those that have a significant economic impact, such as yields, material balance or utility usage. In virtually all practical cases, the model is based on steady state (rather than dynamic) relationships and is non-linear [34]. The choice of measurements to be used for matching the model parameters to the plant operations must be made carefully to avoid an infeasible problem. Bailey et al. [17] chose their variables so that each piece of equipment had enough associated specifications for the available degrees of freedom. Development of process models for optimization is covered in detail by Edgar and Himmelblau [37].

Fatora and Ayala [38] discuss the application of open equation-based models. Instead of posing an equation in its closed form, for example equation (2.9), the equation should be formulated in its open residual format as shown in equation (2.10).

$$Q = mC_p(T_2 - T_1) \tag{2.9}$$

$$R = Q - mC_p(T_2 - T_1) \tag{2.10}$$

A simultaneous equation solving and optimization package will manipulate the unknowns such that the residual terms, R, are driven to zero. This method gives the same results as the closed form equation. The advantage of open form equations is that the model formed from them are more easily maintained than closed form models. Such models are easily modified to account for process changes in the plant since the convergence scheme is separated from the model. The maintenance staff need not recode the convergence scheme each time the model is changed. The open equation-based

approach to process simulation has been used successfully in industry for several years with the proper mathematical tools available.

Two papers by Forbes et al. [43, 44] give some criteria for helping chose a model adequate for real time optimization. The quality of the model has a large effect on the success of the real time optimization system. They give a point-wise model adequacy criterion as the ability of the model to have an optimum that coincides with the true plant optimum. Analytical methods for checking the point-wise model adequacy were developed using reduced space optimization theory, and from these more practical numerical methods were developed for use on industrial problems [44]. A later paper by Forbes and Marlin [42] extends the point-wise model accuracy criteria to yield a set of necessary conditions which must be met for a given real time optimization system to exhibit zero-offset in the manipulated variables with respect to the plant optimum. They also introduce the idea of Design Cost as a framework within which real time optimization design decisions can be made. The Design Cost metric consists of two terms: one for offset from the plant optimum manipulated values and one for the covariance of the model based predictions of the optimal manipulated variables. A systematic method for selecting which parameters should be adjusted by the real time optimizer based on Design Cost is presented.

Bailey et al. [17] briefly cover some modeling issues affecting their nonlinear optimization of a hydrocracker fractionation plant. They found that large equation-based models are sensitive to scaling. However choosing appropriate units avoided numerical problems, even if the resulting units were unusual (e.g., TJ per day). Numerical stability of the model can also be a problem and care should be taken to avoid division as an operator. Also, during the optimization, bounds must be placed on certain variables to prevent the solution from moving too far from the starting point. This is because it takes a finite amount of time to move the plant to a new operating point, therefore the set points should not be moved a great deal on each run of the optimizer. They suggest choosing bounds on the variables to reflect the

allowable change in 1 to 3 hours. The maximum change should depend on the length of time it would take the control system to take the maximum step and bring the plant back to steady state.

A common method for more rapidly computing the optimum is an "inside-out" algorithm first proposed by Boston and Britt [21] to solve a single stage flash algorithm. In this method the major iterations deal with updating the parameters for the simplified models and matching their properties with a rigorous model. Therefore since the more computationally expensive model is only evaluated in the outer loop, considerably less effort is required than with a conventional procedure. However, Biegler et al. [19] and Forbes et al. [44] point out that when this method is applied to optimization problems, this approach based on simplified models can converge to a solution that may not necessarily correspond to the optimum predicted when only rigorous methods are used. They give two observations to help avoid this problem when the "inside-out" type of algorithm is used:

1. A necessary condition for an appropriate simplified model for optimization is that it recognizes the rigorous model optimum as a Karush-Kuhn-Tucker (KKT) point, that is an optimal point.

2. A sufficient condition for an appropriate simplified model is that it matches all the gradients of the rigorous at all points of interest.

Several papers cover the use of commercially available process simulation packages used as the process model, although there is no evidence to date of this method being used in industry. Two types of simulation packages are available. The first are open form, equation oriented or simultaneous process simulation packages, for example, SPEEDUP and MASSBAL. Although these simulators are more flexible, they generally require a high level of expertise to learn and use, have poor user interfaces, and are not widely used in industry. The second type is the closed form or sequential modular process simulation packages, for example, ASPENPLUS, PRO II and HYSIM. These packages are widely used in industry due to their proven reliability and ease

of use, however, they are less efficient with respect to speed and computer processing power required [45]. In addition they require special techniques and insights to develop and change the optimization strategies for on-line operation due to the sequential modular architecture.

Gallier and Kisala [45] used ASPEN, a sequential modular package, in their study of real time optimization. They cover two methods where the simulator is treated as a"black-box" and as an infeasible path optimization using the SQP method. It was found that the latter is preferable as the constraints at each iteration need not be satisfied. This allows the optimization of the process model to be completed in as few as five simulation time equivalents. Macchietto et al. [70] used the equation-based SPEEDUP process simulation package as the process model and also to perform the data reconciliation, gross error detection and optimization steps. Dynamic real time optimization is also possible using the SPEEDUP package. Shewchuk and Morton [93] outline the use of SACDA's MASSBAL process simulation packages as the model in their OPTIMIZER package. This package is also capable of both steady state and dynamic optimization.

2.1.5 Optimization

Objective Function

The objective function is derived from the plant model, and is to be either maximized or minimized through changes in the independent variables. The objective function for real time optimization normally represents the economic model of the process, usually limited to variable effects and ignoring fixed costs such as labour, overheads, etc. A typical objective function given in Darby and White [34] is:

$$\begin{aligned}\text{Objective} = &\text{Product value} - \text{Feed costs} - \text{Utility costs} \\ &+ \text{Other variable economic effects}\end{aligned} \tag{2.11}$$

Bailey et al. [17] cover objective function development in detail. Feed

and product prices must be chosen carefully since different values are often associated with the same material. For example the product might have a contract price, spot market price and also be an intermediate stream sent to another part of the plant for further processing. The case of a stream being an intermediate stream and a product is particularly difficult to determine an accurate value for an inter-process stream, as many planning models only consider feeds and finished products. One remedy for this problem is to include a simple model of the finishing process in the objective function thereby giving an approximate value to the inter-process stream.

Since the optimizer will be estimating derivatives of the objective function with respect to operating variables, it is important that the objective function be continuous. Discontinuities arising from streams with multiple prices should be treated specially (e.g., the stream has a contractual price and a spot price). Often an average price for the stream is not acceptable, and the discontinuity caused needs to be handled by a method such as an integer programming technique.

Bailey et al. [17] also show how they developed their objective function for the real time optimization of a hydrocracker fractionation plant in some detail. Edgar and Himmelblau [37] cover the development of objective functions including methods for measuring profitability and cost estimations.

Optimization Algorithm

The optimization algorithm uses the process model and the objective function to solve for the new optimum set points for the plant. The important considerations in choosing an optimizing algorithm are computational requirements, number of objective function evaluations and robustness. The general form of the constrained optimization problem using the process model

and the objective function [38] is:

$$\begin{aligned} &\min_{z} (\text{or } \max)[F(z,p)] \\ &\text{subject to } f(z,p) = 0 \\ &\qquad z_l \leq z \leq z_u \end{aligned} \tag{2.12}$$

where $F(z,p)$ = the objective function;
$f(z,p)$ = the vector of n equality constraints (from the process model);
p = fixed variables or parameters (e.g., fouling factors, cost coefficients etc., changes in these variables will change the optimum values of z);
z = the process variables which can be varied to optimize F (the number of these variables must be greater than n);
z_l, z_u = the lower and upper bounds on the variables z respectively (which give the inequality constraints on the model).

Methods on how to solve the above constrained optimization problem are covered in several texts [37, 40, 47]. A great deal of literature is available on optimization algorithms, and generally is considered to be a mature area of mathematics.

Presently, the most common type of algorithm reported in the literature for solving this type of problem are Sequential (or Successive) Quadratic Programming (SQP) techniques. The SQP method approximates the objective function by a quadratic function, and the constraints by linear functions, so that quadratic programming can be used recursively to find a search direction minimizing the objective function. At each outer iteration (or quadratic program solution) the SQP method constructs these approximations using information about the values of the variables and their derivatives with respect to the decision variables [37, 40, 69, 93, 103]. Edgar and Himmelblau [37] provide a table of the sources of computer codes for SQP methods. Lucia and Xu [69] provide an extensive review of the SQP algorithms available, particularly with respect to chemical processes and large-scale problems. Fatora

and Ayala report the use of an SQP algorithm in their real time optimization with over 36,000 variables and 31,000 equations.

Bailey et al. [17] use the MINOS routine developed by Stanford. MINOS is a sequential quadratic programming technique which also makes use of the Simplex method and a projected Lagrangian algorithm. The MINOS method is detailed by Murtagh and Saunders [77], and has been shown to solve large problems of around 5000 variables and 4500 equality constraints successfully.

The DICOPT routine is discussed in a paper by Kocis and Grossman [58]. DICOPT is an outer-approximation/equality relaxation algorithm specifically designed for process systems involving mixed-integer nonlinear programming (MINLP) due to nonlinearities in the models. The algorithm was tested successfully using problems of up to 528 variables and 1171 constraints, including 74 binary variables.

2.2 Methods of Real Time Optimization

2.2.1 Global or Centralized Approach to Real Time Optimization

This is the most common approach found in the literature. It involves optimizing the plant as a whole using one objective function, subject to constraints, and a model of the entire plant.

Several examples in the literature use very similar techniques of taking a rigorous steady state model of the process which is then optimized using a nonlinear objective function and constraints. In all the cases listed below, the model appeared to be developed specifically for the real time optimization implementation. Examples of this general form were performed on the following plants:

- Hydrocracker fractionation plant, Bailey et al. [17]. The optimization algorithm was carried out using MINOS 5.1 on nonlinear mass and energy balances models and nonlinear constraints. Comparison is made to

a rank-null space decomposition SQP (Sequential Quadratic Programming) optimization technique. The problem involved 2836 equations, 2891 variables and 10 degrees of freedom. Modeling issues were found to be problems with scaling of the problem, numerical stability, particularly when using the division operator, and minimizing the number of relinearizations of the model constraints required to find the optimum. Bounds were placed on the maximum amount any variable could be changed in a run of the real time optimizer to prevent large set point changes destabilizing the plant. Objective function development was also covered in detail.

- Olefins plant, Fatora and Ayala [38]. Rigorous, fundamental chemical engineering models were developed for the olefins plant. All of the models were written in the open equation form and were optimized using an SQP algorithm. The resulting complex nonlinear optimization problem contained over 36,000 variables and 31,000 equations, and approximately 20 minutes is taken to complete a full cycle of the real time optimizer. The real time optimizer was commissioned in February 1991, and took a total of three weeks to fully implement. The payback was less than one year. The observed benefits from installing the real time optimizer were:
  - Improvements of 5% to 10% of the value added by the process.
  - Consistently holding the process at production targets.
  - Decreased product variability.
  - Optimal handling of utility versus yield trade-off.
  - Many perceived bottlenecks were eliminated.
  - Energy savings.
  - Increased productivity by trending unit performance and monitoring of key parameters.
  - More accurate off-line planning.

- Olefins plant, Lojek and Whitehead [68]. This reference gave very few specifics on what methods they actually used in the real time optimization. They do however outline a very generalized method. The model was based on a set of modular blocks of successive plant sections. The advanced local control techniques used in the plant were also detailed in the paper.

- Hot acid leaching and strong acid leaching residue treatment plant in a zinc refinery, Krishnan et al. [60]. The purpose of this real time optimization was to show the effectiveness of the robust parameter estimation method as outlined in Krishnan et al. [59] and the development of the rigorous steady-state model required. However, the reference mentions very little about the optimization itself.

- Ammonia process, Tsang and Meixell [102]. A rigorous plant model built from first principles was used for the ammonia unit handling 1600 t/day. They found that statistical regression type models were found to have significant disadvantages to the rigorous model approach. The model was performed in Exxon Chemical's Equation Manager and Solver (EMS) package using 5500 equations with 160 tear streams taking two person years' effort to set up.

- Packed-bed immobilized-cell reactor, Hamer and Richenburg [48]. Used a recursive least squares algorithm and high pass filtering of the data to estimate the parameters in the discrete-time dynamic model which was optimized by a full Newton optimization algorithm using the objective function's curvature information. The process was relatively small compared to most of the other plants listed, although it did involve very noisy process data.

- Refinery power plant, Wellons et al. [103]. The MESA nonlinear simulator was selected to model the power plant unit as it was specifically developed for power plants. Mobil's proprietary SQP optimiza-

tion technology, MOPT, was used as the optimization algorithm. They ran the real time optimization algorithm every 30 minutes, but have the option of execution on demand. The system averages 20 updates to the plant per day. From an audit of the on-line power plant optimizer, savings of approximately $2.1 million/year or a 3% saving in fuel consumption were found.

- Pulp refining process, Strand et al. [97]. They found that the model based control technique was dependent on three main factors:

  1. Robust, mechanistic models which accurately describe the effects of process conditions over a wide range of operating conditions.
  2. Tuned parameters for the refiner models so that the equations accurately represent the real process.
  3. Optimization and control techniques which allow the refiner models to be applied in a systematic fashion.

  They found the weaknesses of their optimization system were maintaining the flow of information to the model and ensuring the correct information is available. Maintenance of the system was found to require a dedicated effort from both the mill and vendor.

- Refinery hydrogen plant, Bussani et al. [23]. On-line data reconciliation, gross error detection and optimization were carried out on the hydrogen plant. A sequential modular modelling approach was used. The model contained 27 units and 51 streams, including 8 recycle streams. The optimization was solved using a feasible path/black box method based on SQP. The paper covers the development of the data reconciliation and optimization steps in some detail.

- High-pressure low-density polyethylene (LDPE) tubular reactor, Kiparissides et al. [57]. Parameter estimation, modelling and optimization issues specific to the LDPE tubular reactor are discussed in this paper

in some detail. Unfortunately very few numerical or structural details of the real time optimizer are given. It was found that real time optimization of this process offers large economic gains due to its high volume and extreme operating conditions on one hand, and tight profit margins and quality specifications on the other.

Jang et al. [56] took a two-phase approach to the global real time optimization method. The problem of plant operation is based on the idea of a moving time horizon. The method uses the present measurements and the measurements taken during the past period of time, together with a partial knowledge of the plants physical and chemical principles governing the plant, to give an optimal time plan for the manipulation of the set points for a future period of time. The plant is modeled by either a steady state or dynamic model depending on how stable the plant is. The two phase approach is summarized in Figure 2.7. The identification phase of the method is only carried out when the process has been significantly disturbed from the previous steady state.

Several authors have proposed using process simulation packages for performing real time optimization and data reconciliation of chemical processes:

- Zhang et al. [105] used a combination of ASPEN PLUS for process optimization and parameter estimation, and GAMS/MINOS for data reconciliation and gross error detection to form a real time optimizer for a Monsanto sulphuric acid plant. The data reconciliation and gross error detection method use was the bivariate approach proposed by Tjoa and Biegler [101]. The data reconciliation involved 39 process measurements, 29 of which were required to determine the state of the process. Parameter estimation in ASPEN PLUS used a least squares optimization to determine the 22 key parameters. Other details such as data validation and simulation errors were also covered in this paper. A profit improvement of 17% was achieved, as well as a 25% reduction in stack gas emissions over the previous operating conditions.

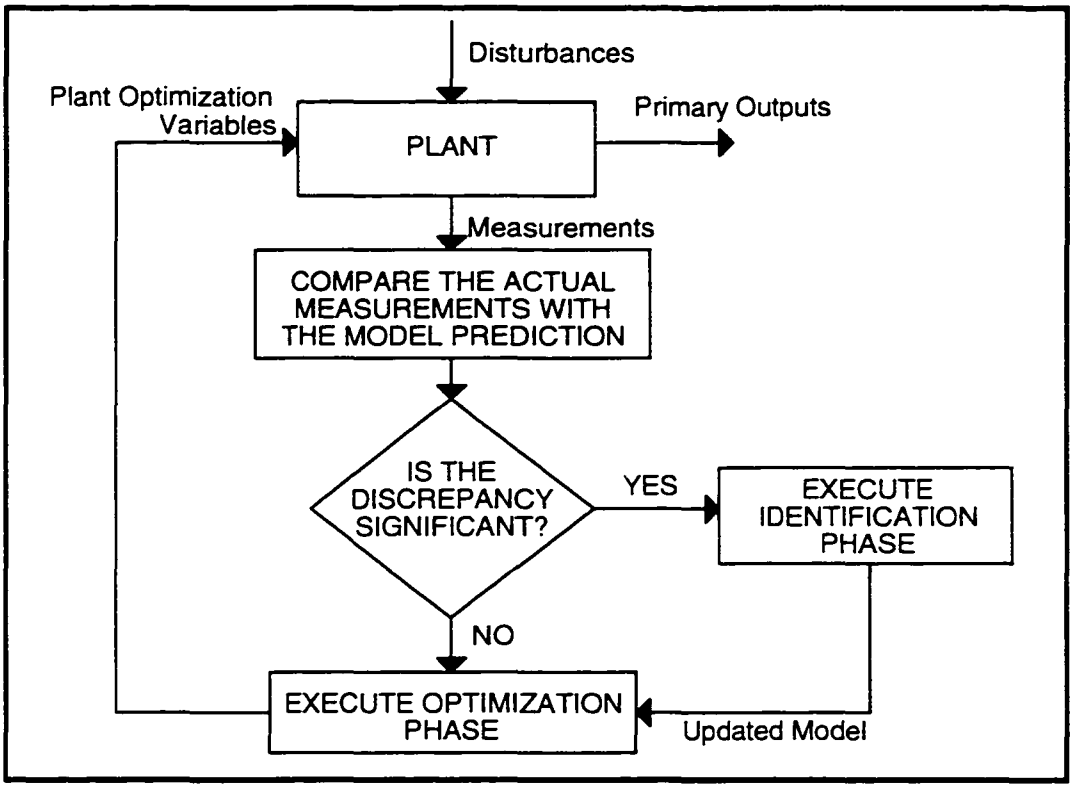


Figure 2.7: Two phase approach to on-line optimization [56].

- Gallier and Kisala [45] use process simulation as the process model using a SQP optimization algorithm to solve for the optimum. Their steps to create the real time optimizer, given the model of the plant, were to:

  1. identify the objective function.
  2. identify the degrees of freedom for the optimization.
  3. identify the constraints.

  They found the SQP method was very efficient at finding the optimum of the ammonia synthesis example they tested. However, they did not extend the method to a real plant situation nor mention the use of data reconciliation or gross error detection in their method.

- Bossen et al. [20] discuss the use of the equation oriented simulator GHEMB for simulation, optimization and data reconciliation of chemical processes. They found that the equation oriented approach was better suited for handling complex and highly integrated process flowsheets. Examples of data reconciliation and optimization were performed on an ammonia process containing 5 recycle streams using a SQP optimization algorithm.

- Macchietto et al. [70] used a an equation oriented simulator, SPEEDUP, a data screening program called DEBIL and the SPEEDUP External Data Interface (EDI) as a communication mechanism to form a real time optimizer. Figure 2.8 gives an outline of plant/control/simulator system. SPEEDUP has an advantage of having both SQP and MINOS optimization routines built into the simulator. They conclude that the SPEEDUP flowsheet optimizing package, using the built in external data interface, can be used successfully on-line with a plant control system. Only case studies are presented in the paper, and no evidence of an industrial application of this method of real time optimization was given.

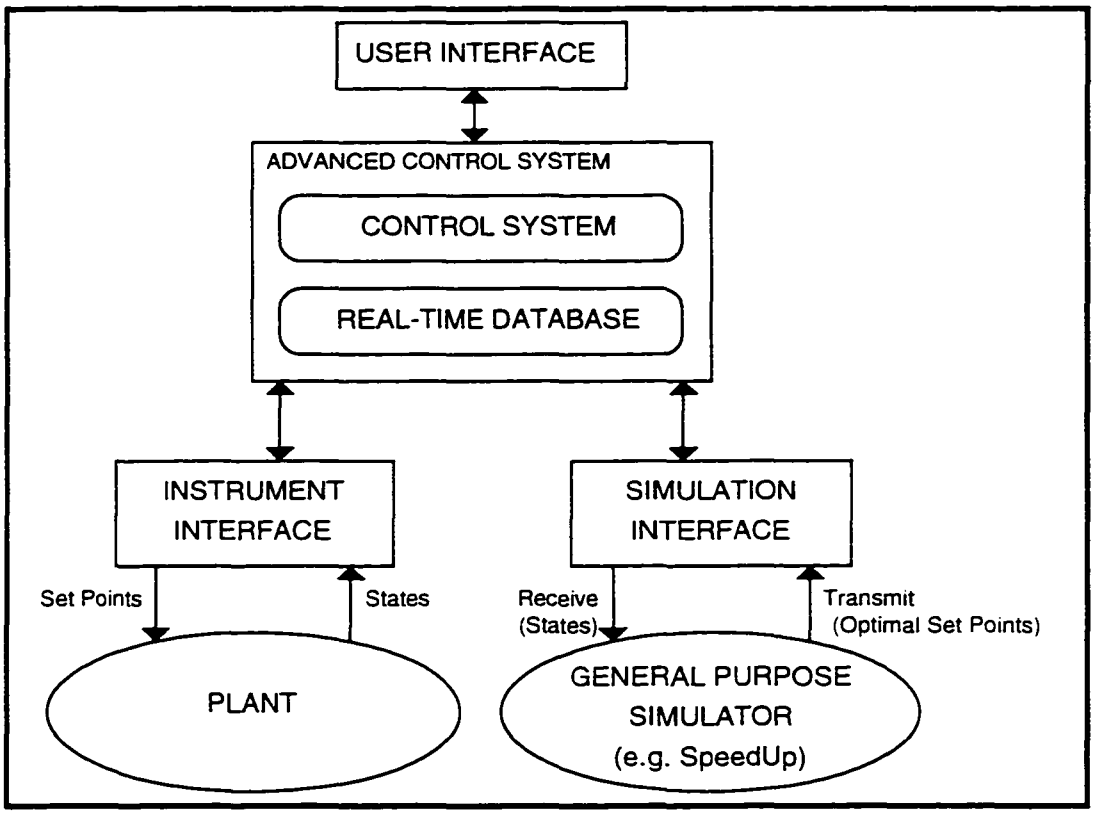


Figure 2.8: Integrated plant/control/simulator system [70].

Several process simulation companies are also developing real time optimization and on-line modelling systems. Honeywell Hi-spec Solutions is developing a system called RTEXEC (previously called OPTIMIZER) based on their equation oriented process simulator MASSBAL [51, 52, 93]. Aspen Technology is also currently developing an equation oriented real time optimization system called RTOPT [14]. Simulation Sciences is developing a rigorous on-line modelling system called ROM which will also be capable of real time optimization [94]. Hyprotech and MDC Technologies are also developing a real time modelling and optimization package called HYSYS.RTO+ [55]. At the time of writing, there were no known industrial applications of these three packages as real time optimizers.

2.2.2 Distributed Approach to Real Time Optimization

The distributed approach breaks down the overall optimization into several local optimizations which are coordinated by a unit optimization/coordination model. Figure 2.9 contrasts the distributed approach to the centralized or global approach [34]. The aim of the distributed approach is to decompose the large scale plant into subsystems thereby reducing the complexity of the original optimization problem [3]. The distributed approach is also sometimes referred to as modular or hierarchical optimization. Distributed real time optimization has been likened to a "bottom-up" implementation of optimization as opposed to the "top-down" centralized approach.

Darby and White [34] give a proposed overall control systems structure to deal with the distributed real time optimization shown in Figure 2.10. Some short examples are given to illustrate that the distributed approach will lead to the same answer as the global approach, but very little detail on implementation of the method is given.

Arkun and Stephanopoulos [3] give a very detailed method of how to perform optimization on a plant decomposed into its subsystems with functional

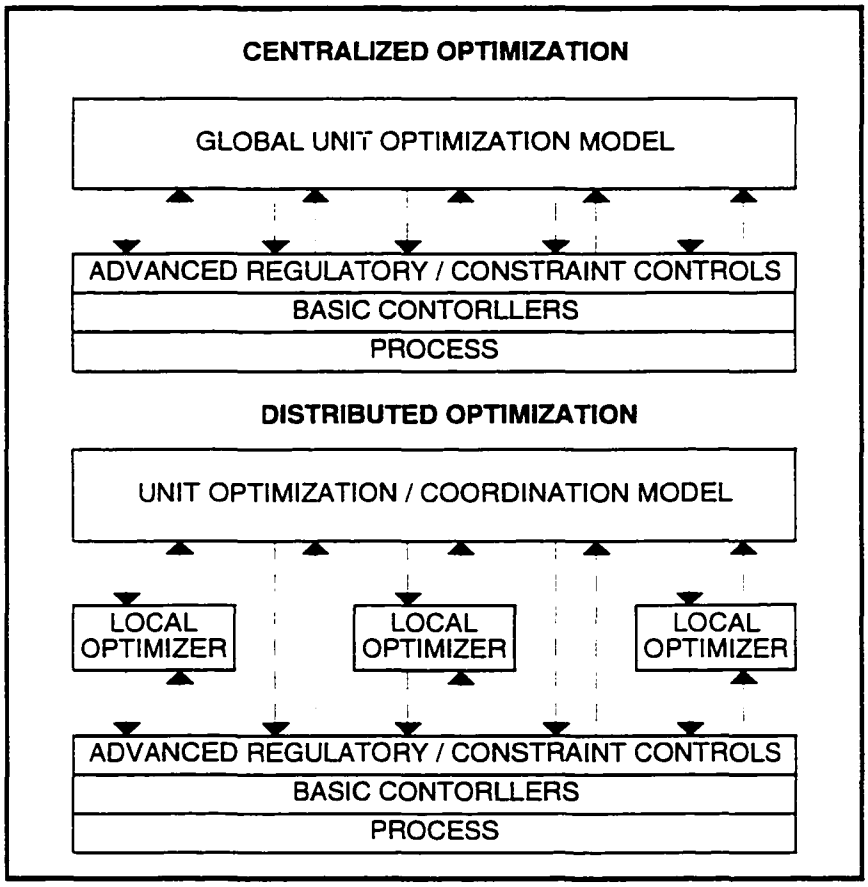


Figure 2.9: Centralized vs. distributed optimization [34].

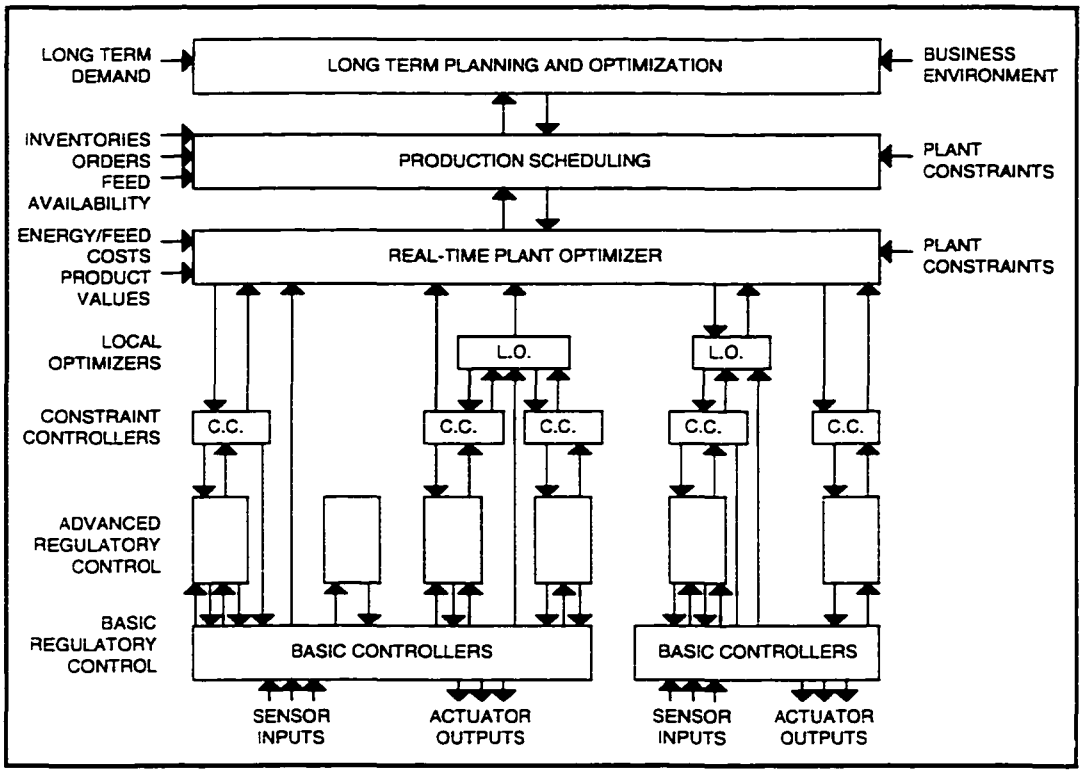


Figure 2.10: Overall process control systems structure [34].

uniformity and common objectives in terms of economics and operation. The subsystems are not independent of each other so interconnections between the subsystems are defined. Local optimizing controllers are designed for each subsystem, S_i, of the plant. If the plant has N subsystems, then the subproblem $S_i, i = 1, \ldots, N$ is subject to the system constraints, as given by:

$$\min_{u_i, m_i} l_i = \Phi_i(x_i, m_i, u_i, d_i) - \lambda_i u_i + \sum_j \lambda_j^T Q_{ji} y_i \tag{2.13}$$

where d_i = vector of disturbances entering the subsystem S_i;
l_i = sub-Langrangian for a subsystem i;
m_i = vector of manipulated variables;
Q_i = incidence matrix denoting the interconnection among the subsystems;
u_i = vector of interconnection points;
x_i = vector of state variables;
y_i = vector of output variables;
λ_i = Lagrange multipliers for the interconnection constraints;
Φ_i = subobjective function.

The coordinator or overall optimizing problem for the plant attempts to satisfy the interaction balance between the subsystems by satisfying:

$$u_i = \sum_{j=1}^{N} Q_{ij} y_j \tag{2.14}$$

By performing this decomposition, and by properly formulating the subsystem control objectives, the selection of the controlled and manipulated variables can be made in a decentralized fashion. The method also deals with the sequencing of the set point changes using "Sequencing Trees" to ensure plant safety and bottlenecking are considered. The best route selected should also try to reduce the number of set point changes made and select the smoother and faster routes with respect to plant operation.

There is no agreement in the literature on whether distributed is better than global real time optimization or not. Darby and White [34] claim the following advantages for the distributed approach:

- The distributed optimization can be carried out more frequently than the global approach as the method only has to wait for steady state in each subsystem rather than the plant as a whole.
- The local optimizers can model the subsystem more accurately and completely than a global optimizer can as the individual model dimensionalities will be less restricted then the global model. Also different models can have different optimization levels.
- Local optimizers permit the incorporation of on-line adaptation or on-line parameter estimation more easily.
- Local optimizers are easier to maintain as they are less complex and hence easier to understand than a large global optimizer.
- If problems are occurring in the modeling, the local optimizer causing the problem can be taken off-line while the rest of the optimizers continue to function. A global optimizer would have to be completely turned off to be restructured.

Bailey et al. [17], however, point out two major difficulties with the distributed approach:

- The constraint information being passed between local unit optimizers to prevent conflicts is not as effective as that in a global approach.
- It is possible to get inconsistencies in the update of the parameters when only parts of the process are considered in the local optimizers.

Unfortunately no actual case studies using the distributed approach were available to help determine which approach gives better results.

2.2.3 Direct Methods (Model free approaches)

The real time optimization is performed directly on the process without explicitly using any model. Direct methods are discussed by Arkun and Stephanopoulos [2] who conclude that the method is too slow and simplistic for on-line optimization. The method is an evolutionary operation, sometimes called an on-line hill-climbing gradient search, where after each trial of set point changes, the objective function is measured at steady state and its sensitivity is used to readjust the set point. Large numbers of set point changes and on-line sensitivity measurements are required to observe the behaviour of the objective function, especially when process noise is present. The method is also slow to reach the optimum as it has to wait for steady state after each set point change. In early attempts at real time optimization, this method was used and criticized for its slowness.

2.3 Implementation

Two papers by Latour [61, 62] cover many of the issues relating to real time optimization from a business objective point of view. A list of their requirements for an on-line optimization project to be successful are:

- Select proper independent variables.
- Formulate model requirements.
- Formulate business objective functions.
- Select optimization algorithms appropriate to the nature of the problem (process and business). Always guarantee a feasible solution. Handle partial equipment failures.
- Specify sensor inputs and manual inputs.

- Define human interface for objectives, defaults, economic parameters, physical limits, laboratory analyses. The system must inform people what it is doing and why.
- Specify the interface with regulatory controls, consider timing, move limits, output sequencing, process dynamics, stability and interactions.
- Specify computer hardware and software.
- Consider procedures for maintaining model fidelity, process and economic.

However, what an on-line optimizer will not achieve when it is implemented into the plant is:

- Determine business objectives of the plant.
- Specify the process and economic models.
- Select criteria for finding and defining the plant optimum.
- Select the best set of independent variables.
- Specify mechanical or safety limits
- Determine market requirements for product quality.
- Verify product values, fuel and feed costs, and production rate limits.

All process models are only approximations of real process behaviour. Therefore as operating conditions in the units shift with time, the model coefficients will require updating. Also new constraints may need to be added or old ones deleted at some point. Whether some model maintenance should be carried out manually or automatically is an important decision to be made in the implementation of the optimizer. Ongoing long term maintenance of the on-line optimizer must be carried out after implementation. There have

been many cases of initial successes which later floundered due to an inadequate number of trained personnel assigned to the long term maintenance of the optimizer. This often results in the real time optimizer falling into disuse.

The complexity of the process model also determines the sophistication of the maintenance required. Not all variables and constraints can be included in the process model. The art of optimization modeling is to select the variables that have the most significant effect on the economics and important constraints, while eliminating those of lesser significance [34].

It is difficult to convey to operators what the optimizer is doing and why. Generally, if the operator does not understand the results, the optimizer will be turned off at the first excuse. Simplicity and modularity in the interface to the operators will enhance understanding [34]. Campbell [24] and Hanmandlu et al. [49] both cover features of on-line optimization software. Figure 2.11 shows how an on-line optimizer interacts with other parts of the plant. Good training and documentation at both the user and system level are required to get the best results out of the optimizer.

The real time optimizer needs to be able to operate without some available process measurements. If the optimization is disabled every time a single input is off-line, then it will not be in use very much in a large plant. Methods for getting around this problem include estimating the missing input or being able to switch off that portion of the optimizer affected by the loss of the input [34].

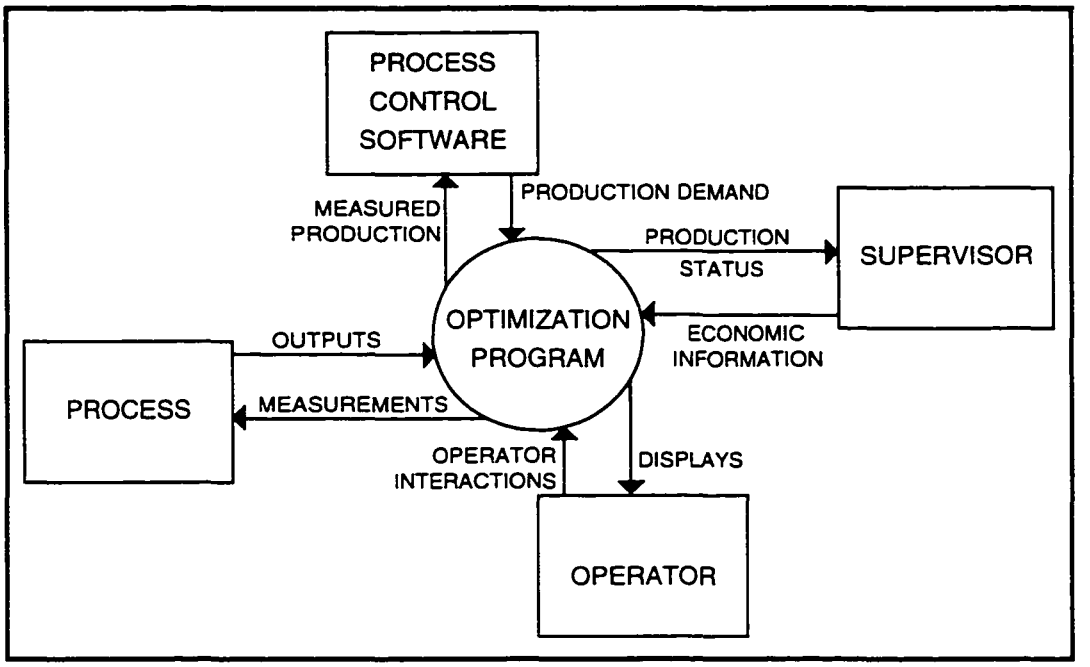


Figure 2.11: Interaction of optimizing program in the plant system [24] [49].

Chapter 3

Process Simulation Model Development

3.1 Introduction

3.1.1 Objectives

The objective in developing the process simulation model is to represent the real stabilizer-splitter process as closely as possible. This process model will be used as the basis for both the data reconciliation of the raw plant data and the subsequent economic optimization of the process set points. Figure 3.1 highlights the part of the real time optimizer to be discussed in this chapter.

3.1.2 Process Description

The process to be modelled consists of a stabilizer column and a splitter column used to prepare the feed stream for a catalytic cracking unit in an oil refinery. A process flow diagram of the plant to be modelled is given in Figure 3.2.

The stabilizer column is fed by a hydrocracked petroleum based stream on stage 14 of a 29 stage column. Light hydrocarbons and other gases are re-

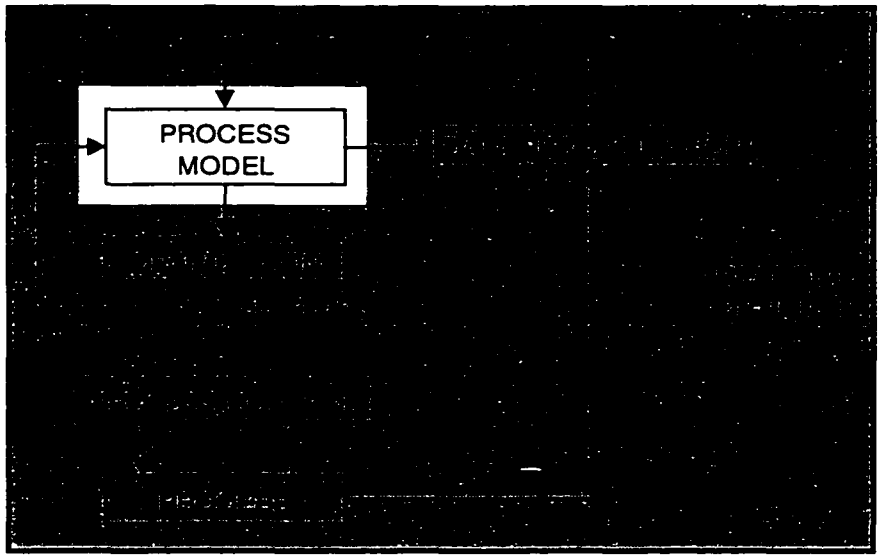


Figure 3.1: Real time optimizer process flow diagram with process model highlighted.

moved in the Vapour product stream from the stabilizer's partial condenser. The liquid product from this condenser, C_4-distillate, contains mainly C_4 hydrocarbons. The bottoms from the stabilizer are passed though a heat exchanger before being fed into the splitter column on stage 21.

The splitter column consists of a 24 stage main column with two 3 stage sidestrippers. Three product streams, Light Iso, Heavy Iso and Light Distillate, are removed from this column, and the bottoms are fed to the hydrocracker. Light Iso is a gasoline product stream removed from the total condenser on the main splitter column. Heavy Iso, also a gasoline product stream, is removed from the reboiler on the first sidestripper. This sidestripper is fed by a liquid draw from stage 15 of the main column, and returns vapour to stage 14. Light Distillate, a heating oil product stream, is removed from the reboiler on the second sidestripper. This sidestripper is fed by a liquid draw from stage 18 of the main column, and returns vapour to stage 17.

Note that the column stages are numbered from top to bottom following

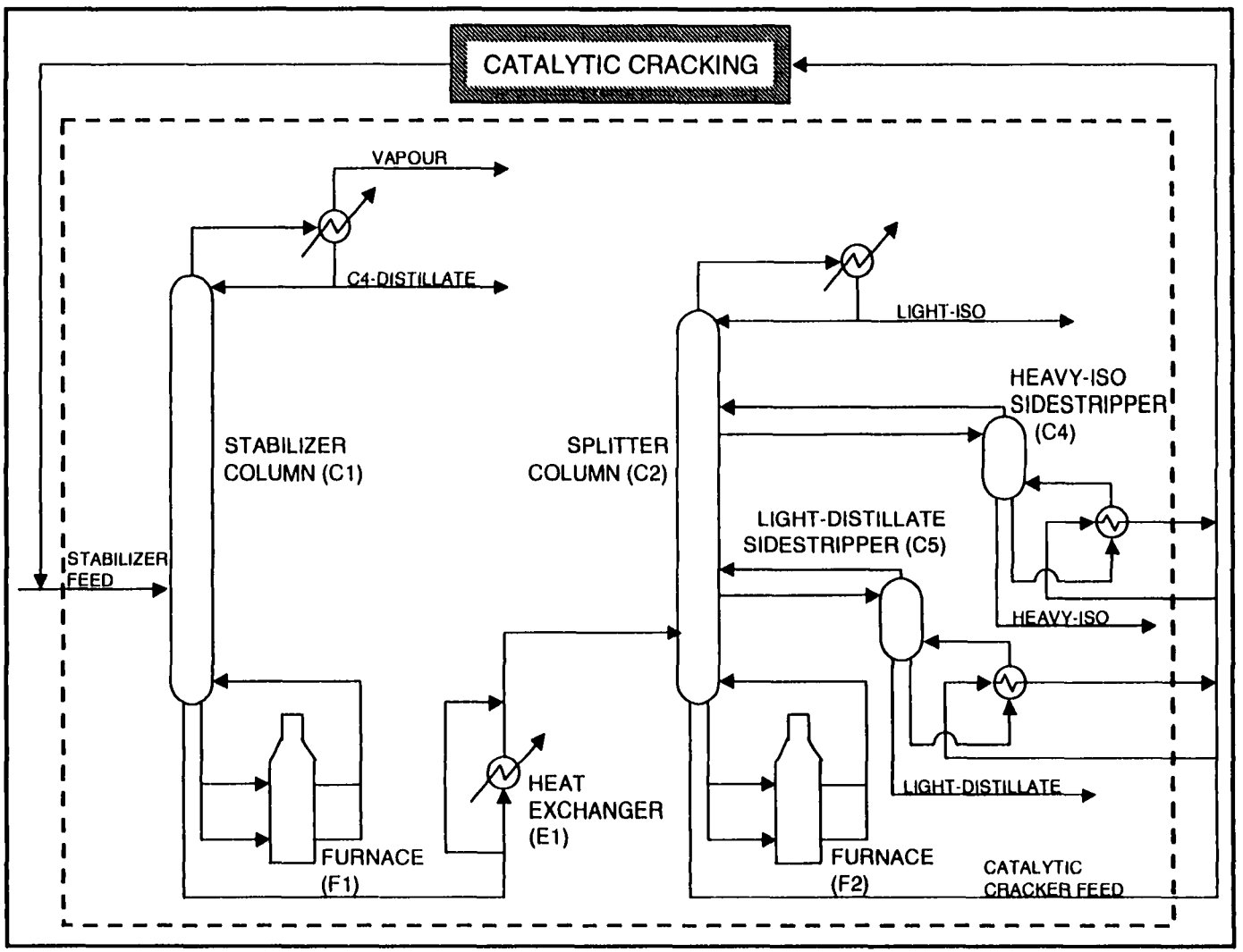


Figure 3.2: Stabilizer-splitter process flow diagram. The portion of the process to be modelled is contained within the dashed box.

the convention used the two process simulation packages used. Also, the stages used in these columns are equilibrium stages, that is, tray efficiencies are not used when modelling this flowsheet.

3.1.3 Development Method

The ASPEN PLUS and SPEEDUP process simulation models were developed using the same method:

1. Choose the components and pseudo-components, and develop the rigorous thermodynamic properties. These were developed based on stream assay data provided by Shell Canada.

2. Develop the process flowsheet around the operating point provided and compare results to those generated by Shell's real time optimizer. The point at which this simulation was run was a typical plant operating point for which the data had already been reconciled to fit mass and energy balances. The results of the simulation being developed were checked against the results of Shell's real time optimization simulation run on the same data.

3. Validate the process model using raw data from the plant. Key variables were checked to ensure that they were being modelled correctly by the process simulation package. A great deal of experience from plant operation was required to check that the results were reasonable and consistent.

3.2 Properties

Both of the process simulation packages, ASPEN PLUS and SPEEDUP, interface with the same properties package, PROPERITIES PLUS [6, 7, 10]. Therefore the properties development for both simulators was identical.

The properties were developed using the MODELMANAGER graphical user interface [8, 9].

Hydrocracked refinery streams typically consist of a range of components from gases such as hydrogen and ammonia to hydrocarbons ranging from methane to nonane and other heavy hydrocarbons. For hydrocarbons components above pentane, many isomers with different properties may exist in the stream. Due to the large number of components present in small quantities, components above pentane are approximated using pseudocomponents. Pseudocomponents are defined in boiling point ranges, e.g., one pseudocomponent may represent all hydrocarbons boiling between 125°C and 145°C. Each pseudocomponent has an associated boiling point, specific gravity and molecular weight which are used to determine other physical and thermodynamic properties for that pseudocomponent. Pseudocomponents are then treated by the simulator in the same way as all other components [4].

The component composition of the stabilizer feed stream was determined from a light ends analysis, true boiling point (TBP) curve, stream specific gravity and stream molecular weight provided by Shell Canada. From this information, the stabilizer feed stream composition was approximated using 7 conventional components and 16 pseudocomponents. The seven conventional components were methane (C_1), ethane (C_2), propane (C_3), *i*-butane (*i*-C_4), *n*-butane (*n*-C_4), *i*-pentane (*i*-C_5) and *n*-pentane (*n*-C_5). The sixteen pseudocomponents were used to represent true boiling point ranges of hydrocarbons larger than *n*-pentane. The naming convention used for pseudocomponents in both simulations is B(T_1)T(T_2) where T_1 and T_2 give the boiling point range in degrees Celsius. For example, B205T235 represents all hydrocarbons with boiling points between 205°C and 235°C.

The Peng-Robinson equation of state is used to calculate all the thermodynamic properties required except for liquid molar volume. Liquid molar volume is calculated for conventional components using the Rackett model. and for pseudocomponents using the API method for molar volume. The Peng-Robinson equation of state is applicable to nonpolar or mildly polar

mixtures such as hydrocarbons including light gases (e.g., carbon dioxide, hydrogen, hydrogen sulphide etc.). This equation of state is recommended for refinery and other hydrocarbon applications [6].

3.3 Sequential Modular Simulation

Development of the sequential-modular process simulation model was performed using the MODELMANAGER graphical user interface [8, 9] to produce an ASPEN PLUS input file. The input file produced is keyword based ASCII text file which can either be edited directly, or imported into MODELMANAGER for editing. Figure 3.3 shows how the process flowsheet given in Figure 3.2 was modelled with ASPEN PLUS.

Five models were required to model the stabilizer-splitter process. The following subsections outline the models used and the flowsheet convergence.

3.3.1 Stabilizer (C1)

The stabilizer column was modelled using the model RADFRAC, a rigorous tray by tray equilibrium based distillation column [5]. Both a vapour product, VAP, and a liquid product, DIST-C4 are drawn from the partial condenser at stage 1 of this column. A liquid bottoms product, SPLTFEED, is drawn from stage 29. The reboiler furnace was simulated as a 2 phase pumparound reboiler with both the draw and return streams on stage 29 of the stabilizer column. To improve the column convergence efficiency and reliability in petroleum applications, the RADFRAC convergence algorithm "SUM-RATES" was chosen.

3.3.2 Heat Exchanger (E1)

The heat exchanger was modelled using the model HEATER, a simple generic heating or cooling model [5]. The preferred model for this block would have been HEATX, a rigorous heat exchanger model, which also calculates the

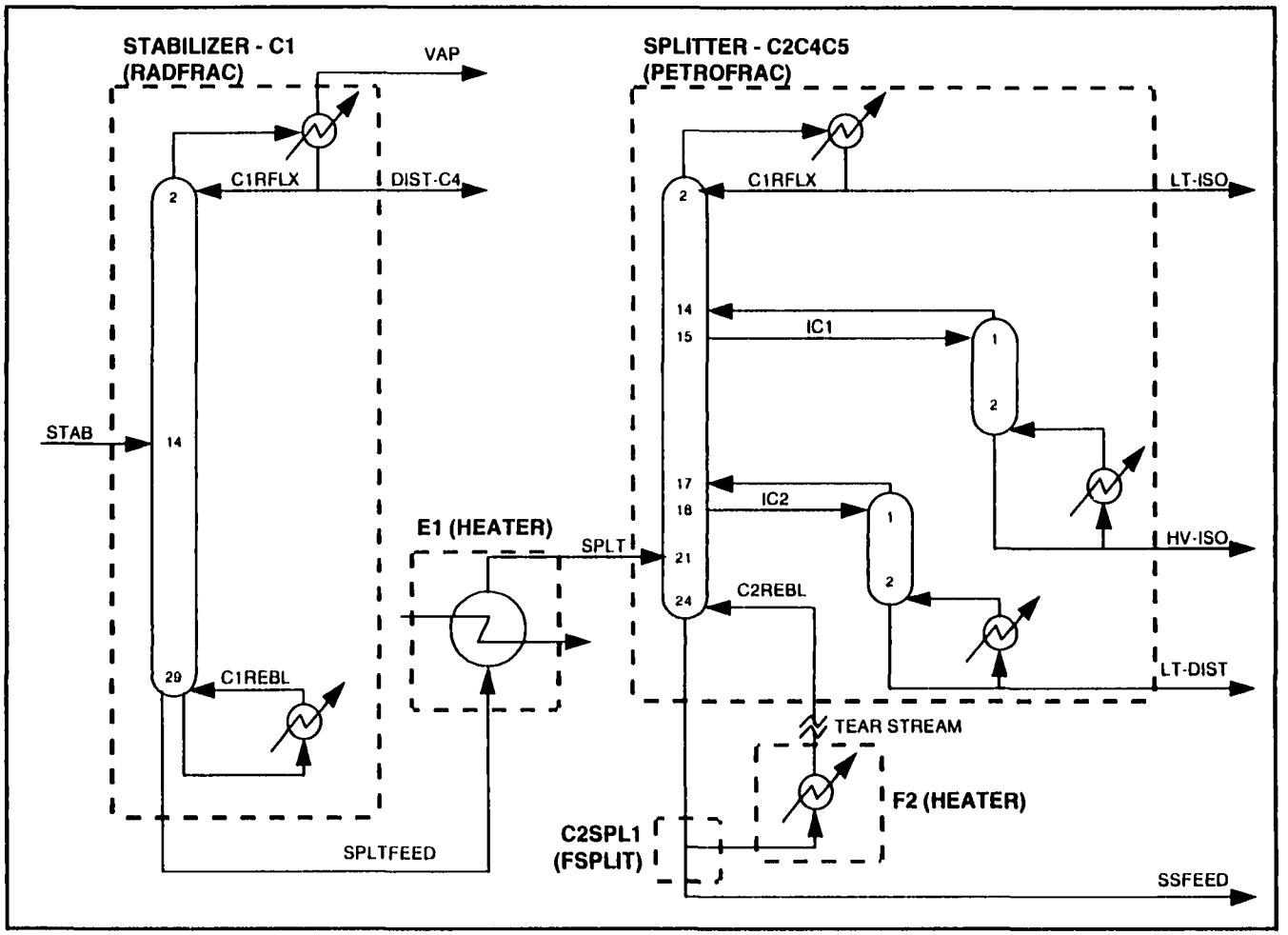


Figure 3.3: ASPEN PLUS stabilizer-splitter process model. The Unit models used are shown as dashed boxes.

heat transfer coefficients. However, there were insufficient measured variables around this unit in the actual process to provide the information required to use HEATX.

3.3.3 Splitter (C2, C4 and C5)

A combination of three models were used to model the splitter: PETROFRAC, FSPLIT and HEATER. The main column, condenser and sidestripper columns were all modelled using PETROFRAC, a rigorous tray by tray equilibrium based distillation column model designed specifically for petroleum applications. Another model, MULTIFRAC, could have been used in place of PETROFRAC to produce similar results, however this is not recommended for petroleum applications [5]. The liquid product stream LT-ISO was drawn from the subcooled total condenser at stage 1. Two reboiled sidestrippers are used to remove the liquid products HV-ISO and LT-DIST. The first sidestripper, C4, is fed by a liquid from stage 15, and returns vapour to stage 14. The second sidestripper, C5, is fed by a liquid drawn from stage 18 and returns vapour to stage 17.

PETROFRAC was unable to model a 2 phase pumparound reboiler. Therefore, the C2 reboiler furnace was modelled externally using a combination of the model FSPLIT, a simple flow splitting block, and a HEATER model. Since these models generate a recycle loop in the flowsheet, a tear stream convergence block was also needed (see Section 3.3.4 below for convergence block details). FSPLIT was used to split the PETROFRAC bottoms into a liquid product stream, SSFEED, and the reboiler feed. The HEATER model was used to reboil the stream C2REBL before it was returned to stage 29 of the main splitter column.

3.3.4 Convergence

A convergence block was required to converge the tear stream created by the external pumparound reboiler on the splitter column. This tear stream was

placed on the stream C2REBL between the reboiler and the main column. The tear stream was converged using the Wegstein method to tolerance, *tol*, of 1×10^{-4} for each variable to be converged using the following expression:

$$-tol \leq \frac{x_{\text{calculated}} - x_{\text{assumed}}}{x_{\text{assumed}}} \leq tol \tag{3.1}$$

This convergence test is bypassed for all components with a mole fraction less than the trace value of $tol/100$. The Wegstein acceleration parameter, q, was allowed to vary between 0 and -5. For a detailed description of the Wegstein method, see either the ASPEN PLUS User Guide [9] or Perry and Green [85]. Converging the tear stream on C2REBL requires iterative solution of the blocks PETROFRAC, FSPLIT and HEATER that make up the splitter column. Typically convergence of the tear stream takes between 15 and 25 passes through these blocks. Initial guesses of the C2REBL stream flow rate, pressure, temperature and composition were also required.

To promote flowsheet convergence speed, estimates of the column temperature, vapour flow and liquid flow profiles were used. These estimates are obtained from a previous run of the process model when it is used in the real time optimizer.

3.4 Equation-oriented Simulation

The equation-oriented process simulation flowsheet was developed by writing a keyword based ASCII text input file. The SPEEDUP Executive is a text based interface which provides syntax checking, specification checking (degrees of freedom analysis) and some tools for run-time error diagnose [11, 12]. Figure 3.4 shows how the process flowsheet given in Figure 3.2 was modelled with SPEEDUP.

Modelling of the stabilizer-splitter flowsheet equipment in SPEEDUP required 7 macros and 24 models. Macros are groups of repeated models used to simplify construction of the flowsheet. In this flowsheet, macros were used to model sections of the distillation columns using multiple distillation tray

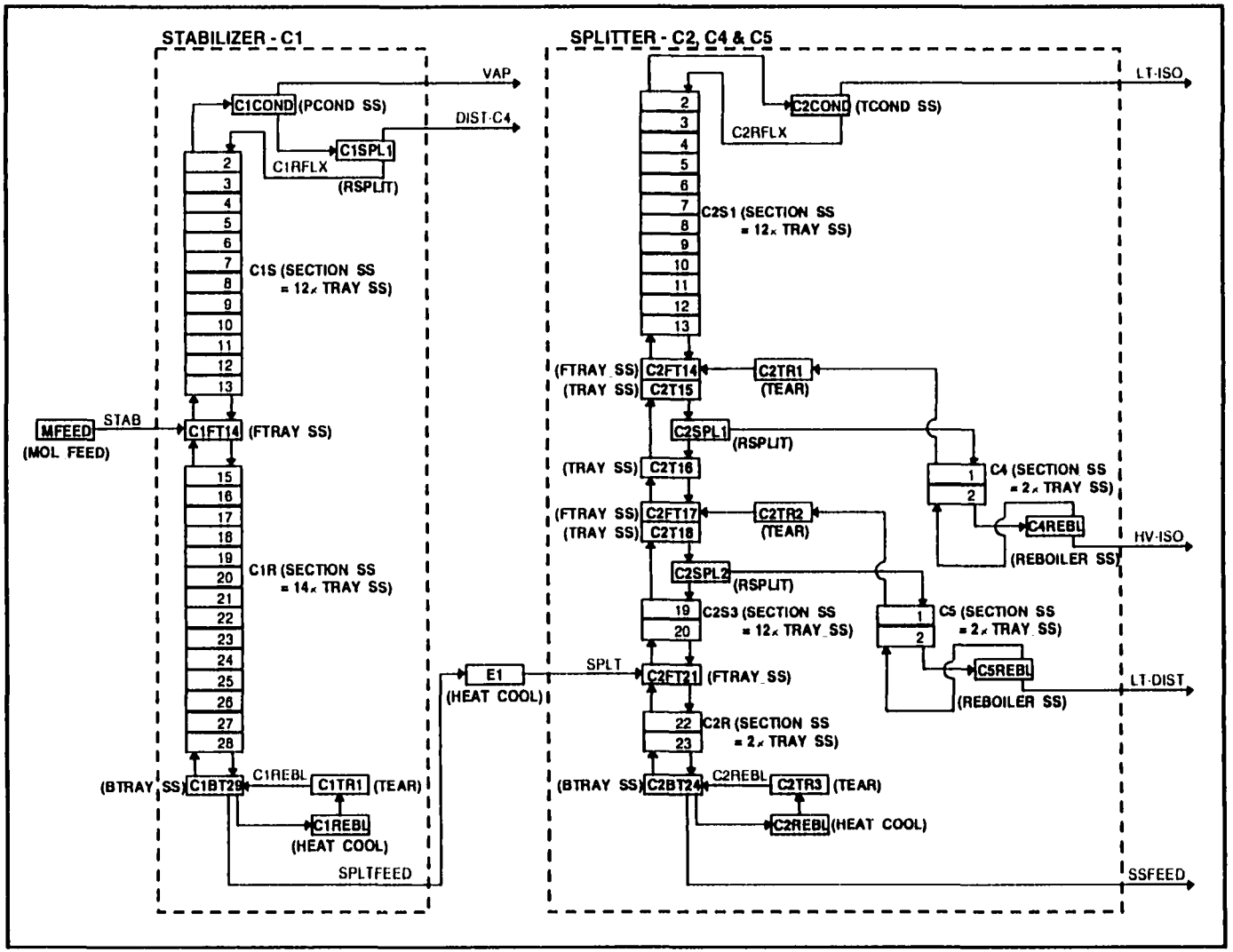


Figure 3.4: SPEEDUP stabilizer-splitter process model. Volume flow and other measurement blocks are not shown.

models. One user defined equipment model was used in the flowsheet as the bottom stage of columns C1 and C2 (see Section 3.4.1 below for details of the model).

In addition to the equipment models, an extra 17 user defined measurement models were required to calculate volume flow rates, C_5 compositions, true boiling point temperatures and Reid vapour pressures (see Section 3.4.2 below for details of these models). Including the models used in the macros, a total of 74 models were used to simulate the stabilizer-splitter flowsheet. The following subsections outline how these macros and models were used, and the flowsheet convergence.

3.4.1 User Defined Equipment Models

SPEEDUP allows user defined models to be added to the flowsheet easily. In this simulation, one user written equipment model, BTRAY_SS was used to simulate the bottom tray of columns C1 and C2. BTRAY_SS is based on the SPEEDUP Library model FTRAY_SS without a vapour inlet stream and two liquid outlet streams. This model was required because the bottom trays of columns C1 and C2 are fed by mixed vapour-liquid streams instead of vapour only inlet streams. Also, two liquid outlet streams are drawn from the bottom tray; the reboiler feed stream and the bottoms product stream.

This model was created by copying the model FTRAY_SS from the SPEEDUP Library and editing the input code. The input code (including the modelling equations) for BTRAY_SS is listed in Appendix E, Section E.8.

3.4.2 User Defined Measurement Models

Seven user defined models were created to measure the various stream properties required by the real time optimizer. These measurement models do not change or affect the flowsheet in any other way. They are included in the flowsheet in the same way as other unit models. The input code for the models outlined below is included in Appendix E, Section E.8.

Volume Flow Rates

Three models, VFLOW, LIQ_VFLOW and VAP_VFLOW, were created to measure standard and true volume flow rates for mixed phase, liquid and vapour streams respectively.

The true volume is calculated by making a property procedure call for the liquid or vapour molar density at the current pressure and temperature of the stream. The volume flow rate is then obtained by solving for the unknown volume flow F_v from the known molar flow F and molar density ρ:

$$F = F_v \rho \tag{3.2}$$

For the model VFLOW, the property calls are made separately for the liquid and vapour phases. The liquid and vapour volume flow rates are calculated separately, and then summed to give the overall stream volume flow rate.

The standard liquid volume flow rate is calculated in the same way as true volume flow, except the property procedure call for molar density is only made for the liquid phase at 60°F and 1 atmosphere. The model VAP_VFLOW also calculates a standard vapour flow using a property procedure call for molar density for the vapour phase at 60°F and 1 atmosphere.

C_5 Composition

The model C5_VFLOW is used to calculate both volume flow rates and the stream C_5 composition. The C_5 composition, `C5S`, is calculated simply by adding the *i*-pentane and *n*-pentane compositions together. This model is identical to LIQ_VFLOW with addition of the calculation for `C5S`.

True Boiling Point Temperatures

The model TBP is used to calculate both volume flow rates and true boiling point temperatures. The true boiling point temperatures are calculated at both 10% and 90% standard liquid volume distilled. A user supplied FORTRAN procedure, which linearly interpolates between the boiling point

temperatures of the pseudocomponent, is used to determine the true boiling point temperatures at both 10% and 90% of the streams standard liquid volume. This model is the same as LIQ_VFLOW with addition of the true boiling point procedure calls.

Reid Vapour Pressure

Unfortunately SPEEDUP does not include a built in procedure for determining the Reid vapour pressure of a stream. Therefore, two user-defined models, M2R and RVP, were created to determine Reid vapour pressure based on the methods outlined in the API Technical Data Book [1] and ASTM standard D323 [41].

The model M2R changes the stream equation of state from Peng-Robinson to ideal, and adds air as the 24th stream component. The model RVP follows the procedure outlined below with the outlet stream of M2R as the inlet stream:

1. Saturate the stream with air at 32°F.
2. Calculate the moles of air required to mix air with the stream at a ratio of 4:1 at 32°F.
3. Flash the 4:1 mixture of air and the stream at 100°F under constant volume. This pressure of this flash under these conditions gives the Reid vapour pressure of the stream.

The model RVP requires that the ideal molar density of air at 32°F and 100°F are set in the simulation operation section. This model was tested against Reid vapour pressure results produced in ASPEN PLUS simulations at several operating conditions and was found to be in close agreement (±1.0%).

3.4.3 Stabilizer (C1)

The stabilizer was modelled using 2 macros and 7 other models. The feed stream, STAB, was specified using MOL_FEED, a molar feed model [10].

The stabilizer column stages were modelled using two SECTION_SS macros and a FTRAY_SS rigorous feed stage model at stage 15. The SECTION_SS macros form both the stripping and rectifying sections of the column, and are made up of 14 and 12 stages respectively. Each of the stages within the macro use the model TRAY_SS, a rigorous distillation stage, which flashes the mixture of the inlet vapour and liquid streams to yield the outlet vapour and liquid steams.

The partial condenser was modelled using PCOND_SS, a rigorous partial condenser model, and RSPLIT, a liquid flow splitting model. The vapour product, VAP, was drawn from the partial condenser. The RSPLIT model divides the condensed overhead liquid into a reflux stream and the liquid product stream DIST-C4.

The reboiler and bottom stage (stage 29) of the column were modelled using the models BTRAY_SS, HEAT_COOL and TEAR. One of the liquid outlet streams from BTRAY_SS was fed to the reboiler, while the other forms the bottoms product. The reboiler was modelled by a HEAT_COOL model, a simple generic heating or cooling model. Since the pumparound reboiler creates a flowsheet recycle loop, a TEAR block was required. The TEAR block was placed on the stream from the reboiler feeding the bottom stage of the column. See Section 3.4.6 for details and convergence of the TEAR block.

3.4.4 Heat Exchanger (E1)

The heat exchanger was modelled using the model HEAT_COOL. Unfortunately a more rigorous heat exchanger model could not be used to calculate heat transfer coefficients due to insufficient measured data around this unit in the actual process.

3.4.5 Splitter (C2, C4 and C5)

The splitter columns were modelled using 5 macros and 16 other models. The main column was formed using 3 feed stage models, FTRAY_SS, 3 standard stage models, TRAY_SS, 2 liquid flow splitting models, RSPLIT, and 3 column section macros, SECTION_SS. The three feed stage models were used at stages 14, 17 and 21. Stage 21 receives the column feed from the heat exchanger E1. Stages 14 and 17 receive vapour phase feed from the sidestripper columns C4 and C5 respectively. The two RSPLIT models are used to the liquid feeds for the sidestripper columns C4 and C5 from stages 15 and 18 respectively. The 19 stage rectifying section of the C2 column is formed by 2 SECTION_SS macros, 3 TRAY_SS and 2 FTRAY_SS models. The stripping section of the column is formed using a 2 stage SECTION_SS macro.

The sub-cooled condenser was modelled using TCOND_SS, a rigorous total condenser model. Both the sub-cooled liquid product, LT-ISO, and the reflux are drawn from the total condenser. The reboiler and bottom stage (stage 24) of the column were modelled with the same models and layout used in the stabilizer column, C1.

The two sidestripper columns, C4 and C5 were formed using the same simulation models. The columns were modelled using 2 stage SECTION_SS macros. The reboilers were simulated using REBOILER_SS, a rigorous kettle reboiler model. A liquid product was removed from the reboilers and vapour was returned to the sidestripper columns. Since the sidestrippers create a flowsheet recycle loop, a TEAR block was placed on the vapour stream being returned to the main column, C2.

3.4.6 Convergence

The process simulation flowsheet in SPEEDUP was solved as a large set of simultaneous equations. After translation of the input code, block decomposition of the flowsheet is performed. For this simulation, 1164 equation blocks used to solve the 6066 equations and unknown variables. These equation

blocks are formed based on the equations in the models used, and the variables set in the operation section of the input code. Only 6 of the 1164 blocks generated are nonlinear and contain more than one equation. SPEEDUP then generates the derivative code for the simulation (13,653 derivative expressions were required). The simulation is then ready for solution.

The linear equation blocks are solved by direct substitution. The 6 nonlinear blocks are solved using a Newton trust region method using a dogleg step (see Edgar and Himmelblau [37], Fletcher [40] or Gill et al. [47] for details of this method). The flowsheet is converged to an accuracy of 1×10^{-5}, that is, all the equation residuals are less than 1×10^{-5}. The residual of an equation is the difference between and left and right hand sides of the equation.

To promote flowsheet convergence, all variables not "set" in the operation section were given preset values. The values used to preset the first runs of this process model were generated by an ASPEN PLUS run of the same model to attain convergence. Later runs used presets generated from previous SPEEDUP simulations to ensure flowsheet convergence.

Tear streams in an equation-oriented simulators do not require a separate convergence scheme from the rest of the flowsheet. In SPEEDUP, a TEAR block is used to add slack variables to the recycle stream which allow for the convergence of the recycle loop. The tear model has equations for flow, temperature, pressure and each composition of the form:

$$F_{in} = F_{out} - F_{slack} \tag{3.3}$$

All the slack variables (e.g., F_{slack}) are set to zero in the input file operation section and the tear stream equations are converged using the same method as the rest of the flowsheet. Therefore, the addition of a TEAR block in this simulation means the addition of 26 equations and variables to the overall flowsheet problem. Four TEAR blocks were required for this flowsheet: two were used in the reboilers for columns C1 and C2, and two were used on the vapour returns from the two sidestrippers C4 and C5.

3.5 Discussion

The following subsections discuss various problems and modelling aspects encountered using both the sequential modular and equation-oriented process simulators. Property development issues are discussed separately as these were common to both simulation packages. A summary comparing various quantitative aspects of the process model simulation development and performance are presented in Table 3.1.

3.5.1 Properties

Property development for both of the process simulation packages took 104 hours to complete due to difficulties in matching the pseudocomponent properties to those of a hydrocracked petroleum stream. Specifically, the properties of the first two pseudocomponents, B45T65 and B65T85 (representing hydrocarbons with boiling points from 45°C to 65°C and 65°C to 85°C respectively), were the most difficult to match. This was because the properties of these components greatly affect the operation of the condensers on both columns C1 and C2. It was found with the original assay data provided that if the correct temperatures were met in the condensers of both columns, the compositions and product flows of the C_4-Distillate and Light Iso streams were incorrect by up to 200%.

The cause of this problem was found to be the assumptions made by PROPERTIES PLUS when determining the specific gravities and molecular weights for each pseudocomponent from the data provided. Since only an overall specific gravity and molecular weight had been given in the assay analysis, PROPERTIES PLUS determines the pseudocomponent properties based on a typical hydrocarbon stream's specific gravity and molecular weight curves. A hydrocracked hydrocarbon stream, however, exhibits different typical property curves due to the presence of a different component mix, particularly at the ends of the curves (i.e., the light and heavy components). Using typical hydrocracked petroleum stream curves, the pseudocomponents were

Table 3.1: Comparison of both process simulators for model development.

	Sequential Modular Simulator	**Equation-oriented Simulator**
Development Time (hours)	192	310
(excluding property development)	(88)	(206)
Input file size (bytes)	10,855	263,366
(without preset variables)	(8,085)	(44,324)
Number of process models used	5	74
Number of tear streams	1	4
CPU time (mins)[a]	1.17	1.51
Stabilizer column flowsheet passes	1	3
Splitter column flowsheet passes	5	2
Largest discrepancy from Shell process model	0.30%	2.65%

[a]CPU time is measured for calculations only on an IBM RS 6000 Model 530H computer.

recalculated. These new pseudocomponents provided very accurate results to within 0.5% of Shell's simulated results.

3.5.2 Sequential Modular Simulation

The initial building and debugging of the process flowsheet in ASPEN PLUS was relatively straightforward. The major problems encountered were during the modelling of the two phase pumparound reboiler on the splitter column using the rigorous multiple distillation column models available, and drying up of the main splitter column during flowsheet convergence.

The model PETROFRAC was a very powerful way to model a column including sidestrippers by removing the need for flowsheet tear streams. PETROFRAC solves the recycle loops created internally within the block, thereby ensuring robust convergence and faster solution. However, PETROFRAC was not as flexible as RADFRAC in simulating the column C2 furnace reboiler. The pumparound model built into PETROFRAC is only capable handling single phase streams. Therefore, the reboiler had to be modelled externally using an FSPLIT and a HEATER model, thereby creating a recycle loop. The addition of this tear stream significantly slows convergence of the flowsheet due to the extra flowsheet passes required to converge the recycle loop. The addition of this recycle loop illustrates the effect of recycle loops on convergence of sequential modular process flowsheets.

Internal convergence of the PETROFRAC block was initially quite difficult. The main column tended to dry up around the stages where liquid was drawn to feed the sidestripper columns. This problem was fixed by adding temperature, liquid flow and vapour flow column profile estimates. Initially these estimates were guesses until convergence of the column was achieved. From then on, the column was initialized using estimates provided from a previous run of the process model. Correction of the pseudocomponent properties as discussed in Section 3.5.1 also helped this convergence problem.

Table 3.2 summarises the testing of the final ASPEN PLUS process model simulation against Shell's RTO process model. From these results and testing

performed by Shell against actual plant data, it was felt that this model adequately reflects the true process for real time optimization.

3.5.3 Equation-oriented Simulation

The initial building and syntax debugging of the SPEEDUP flowsheet input code was relatively straightforward. Obtaining flowsheet convergence, however, was very difficult for several reasons outlined below.

In building the flowsheet, several user written models were required to supplement those in the SPEEDUP model library (see Sections 3.4.1 and 3.4.2 for details). These models were easy to write and implement with the exception of the Reid vapour pressure measurement model. This model required a complex series of flashes and a different equation of state to the rest of the process flowsheet to calculate the Reid vapour pressure. It is probable that in future versions of SPEEDUP, procedure calls to PROPERTIES PLUS will be implemented for Reid vapour pressure and other petroleum specific properties.

The main problem with flowsheet convergence was caused by the large numbers of equations and variables created because of the 23 components in each stream. This resulted in a problem involving 6066 equations and unknown variables which were difficult to analyse due to number of composition variables. Also, a large percentage of these composition variables were at or close to zero. Therefore bound checking was very difficult to perform since usually over 1000 composition variables would be on their lower bound. Bounds on variables in an equation-oriented simulator are used to prevent impossible solutions, e.g., negative compositions, temperatures below absolute zero etc. Bounds can also be used to promote flowsheet convergence by preventing excessive movements in some variables which can prevent solution. However, bounds can also hinder flowsheet convergence if they restrict the movement of the equation solver too much.

Initially, flowsheet convergence was promoted by using a combination of checking the bounds after solution failure and checking the maximum resid-

Table 3.2: Discrepancies between the ASPEN PLUS process model and Shell's RTO process model.

Stream Name	Molar Flow	Temperature
Vapour	0.00%	0.01%
C_4-distillate	0.00%	0.01%
Splitter Feed	0.00%	0.00%
Light Iso	0.01%	0.01%
Heavy Iso	0.00%	0.23%
Light Distillate	0.01%	0.30%
SS Feed	0.01%	0.00%

uals at the end of each iteration. When variables other than compositions hit their bounds, these bounds were widened. For variables with the highest residuals on the first iteration, estimates of the final values of those variables were provided from the results of an ASPEN PLUS process simulation in the form of "presets". Eventually it was found that so many variables required estimates, it was decided to preset all variables using ASPEN PLUS process model results. Flowsheet convergence was then obtained. Future process model simulations were then run using presets of a previous simulation which had converged, using SPEEDUP's ability to save results. These saved results can then be used to preset future simulations through the "Use" sub-environment.

This difficulty with flowsheet convergence shows that for very large, complex simulations using an equation-oriented process simulator, the initial values of the unknown variables dramatically affects the simulator's ability to obtain a solution. In this case, the simplest way to provide estimates was to use the results of a sequential modular simulation. In future runs, it was found that if the process model was initialized using a feasible solution, convergence was usually obtained, even for quite large movements in operating conditions.

Another tool that was useful in promoting flowsheet convergence was

scaling of the flowsheet variables and equations. In this problem, variables ranged in magnitude from 10^{-2} for compositions to 10^{4} for flows. When scaling is used, the flowsheet is solved in terms of scaled variables. A scaled variable is equal to the unscaled variable divided by a user set scaling factor. The aim of scaling is to have all the scaled variables of similar magnitude to promote convergence and performance of the simulation. It was found that scaling did help convergence considerably, however presets were still required for most variables. Unfortunately, scaling has not been implemented in SPEEDUP for use in optimization runs at this time. Therefore the process model had to be able to converge without scaling.

Table 3.3 summarises the testing of the final SPEEDUP process model simulation against Shell's RTO process model. From these results and testing performed by Shell against plant data, it was felt that this model adequately reflects the true process for real time optimization.

Table 3.3: Discrepancies between the SPEEDUP process model and Shell's RTO process model.

Stream Name	**Molar Flow**	**Temperature**
Vapour	0.00%	0.62%
C_4-distillate	0.00%	0.62%
Splitter Feed	0.29%	0.00%
Light Iso	0.00%	0.00%
Heavy Iso	0.00%	2.65%
Light Distillate	0.00%	0.59%
SS Feed	0.67%	0.10%

Chapter 4

Data Reconciliation Development

4.1 Introduction

4.1.1 Objectives

The objective of data reconciliation is to adjust the measured process data to satisfy mass and energy balances. The adjustments made to the process data should be as small as possible to ensure that the reconciled data still accurately represents the true operating conditions of the process. The data reconciliation is to be performed using the process model developed in the previous chapter. Figure 4.1 highlights the part of the real time optimizer to be developed in this chapter.

4.1.2 Data Reconciliation Description

The stabilizer-splitter data reconciliation problem is a nonlinear, steady state, unconstrained data reconciliation involving 28 measurements. Figure 4.2 shows the measured temperature and flow variables available for the stabilizer-splitter process. All the variables shown in Figure 4.2 were reconciled except

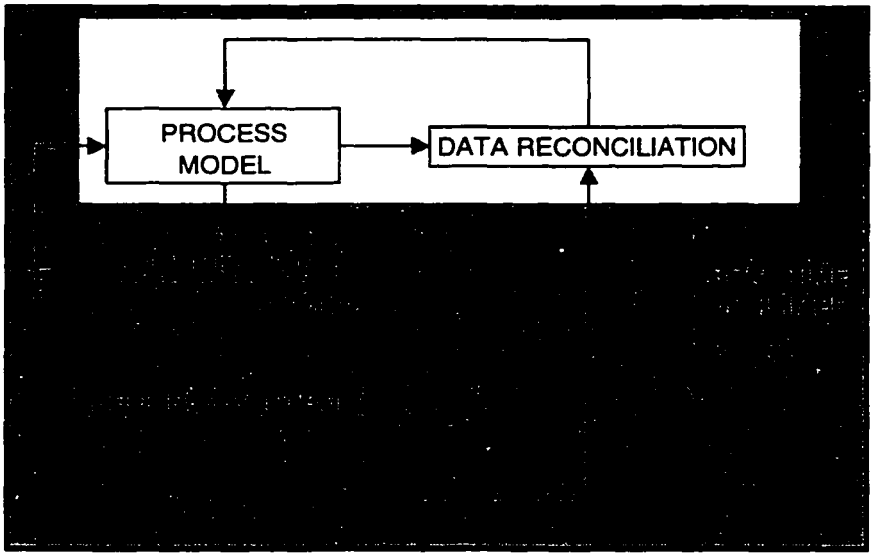


Figure 4.1: Real time optimizer process flow diagram with data reconciliation highlighted.

for the stage 3 temperature, T11. The stage 3 temperature was not reconciled because the process model developed uses equilibrium stages to model the stabilizer column, therefore an accurate measurement of the true tray 3 temperature cannot be made using this process model. See Table B.2 or Table C.2 for full explantations of the measured variables by tag name. No gross error detection or parameter estimation was performed using the process data.

The data reconciliation problem was solved using a weighted least squares objective function similar to the one used by Piccolo [87]. The objective function for this problem is given by:

$$\begin{aligned} &\min_{\mathbf{x}} \left[\sum_{i=1}^{28} E_i \right] \\ &\text{subject to } h(\mathbf{x}) = \mathbf{0} \\ &\text{where } E_i = w_i \left(\frac{y_i - x_i}{s_i} \right)^2 \end{aligned} \tag{4.1}$$

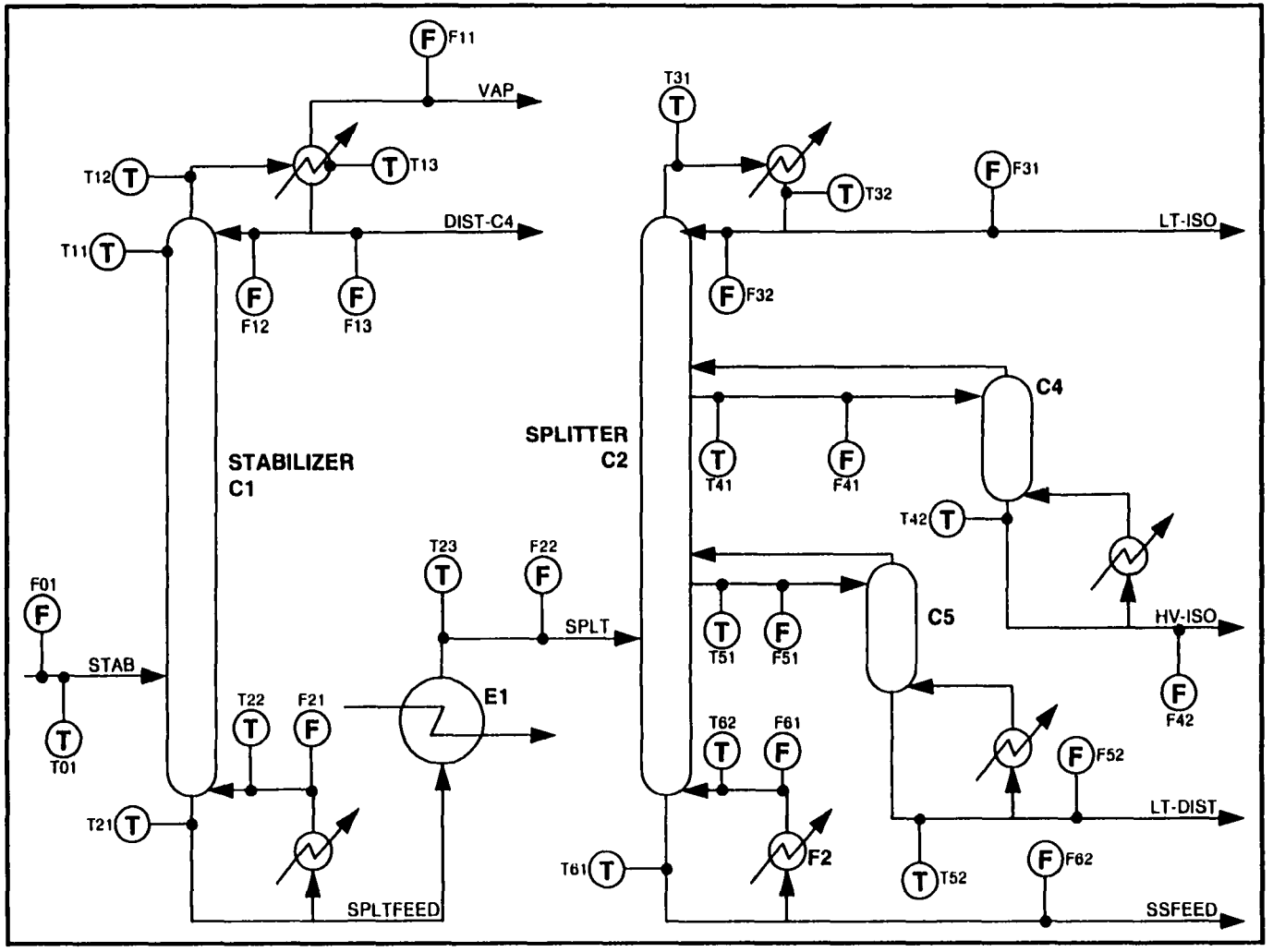


Figure 4.2: Stabilizer-splitter process flowsheet including measurement variable tag numbers.

where E = least squares error term associated with each measured variable:
w = weighting factor associated with each measured variable;
y = the measured variable;
x = the reconciled variable;
s = the standard deviation of the measurement transmitter;
$h(\mathbf{x})$ = the set of equality constraints represented by the process model.

The standard deviations of the measurement transmitters used in the objective function were obtained from Shell. The weighting factors can be used to place emphasis on certain measured variables for a variety of reasons. Weighting of the objective function was implemented in both simulators. however it was not used to place emphasis on any variables in this study (i.e., all the weights were set to same value).

4.2 Sequential Modular Simulation

Development of the sequential modular data reconciliation simulation was performed using the MODELMANAGER graphical user interface [8. 9] to produce an ASPEN PLUS input file. Some direct editing of the input file was also required, particularly in developing the in-line FORTRAN blocks and the FORTRAN objective function. Details of the optimization algorithm used, objective function and manipulated variable selection are covered in the following subsections.

4.2.1 Optimization Algorithm

The optimization algorithm used to converge the data reconciliation objective function was the SQP (sequential or successive quadratic programming) method [9]. This optimization algorithm follows a feasible path for this problem, converging the tear stream at every iteration of the optimization. The exact steps used by the ASPEN PLUS SQP algorithm are not detailed in

any of the manuals. The SQP optimization algorithm is probably based on the general method summarised in Figure 4.3 [37].

The solution of the optimization problem is performed to an optimization tolerance of 1×10^{-1}. That is, until the improvement of the objective function on successive iterations is reduced below this tolerance. The convergence of the process flowsheet and tear streams are performed using the same tolerances and methods detailed in Section 3.3.4.

4.2.2 Objective Function

The objective function in ASPEN PLUS is supplied as FORTRAN 77 code. The objective function given in Equation (4.1) was implemented by calculating the 28 least square error terms, E_i corresponding to each measured variable, first. An example error term calculation for the stabilizer feed flow, F01, is given below:

```
F       EF01 = WF01*((F01-SF01)/VF01)**2
```

where EF01 = least squares error term corresponding to F01;
WF01 = weighting factor associated with F01;
F01 = reconciled value;
SF01 = measured value of F01;
VF01 = the standard deviation associated with F01.

These error terms were then summed to give the data reconciliation objective to be minimized.

The weighting factors, measured values and standard deviations are read from the data files IWGHT.DAT, IMEAS.DAT and IVARNC.DAT respectively using an in-line FORTRAN block (see Appendix B for a detailed description of these data files). This in-line FORTRAN block is executed before the data reconciliation optimization commences. The data read in from the files are stored in FORTRAN "common" blocks for use during the calculation

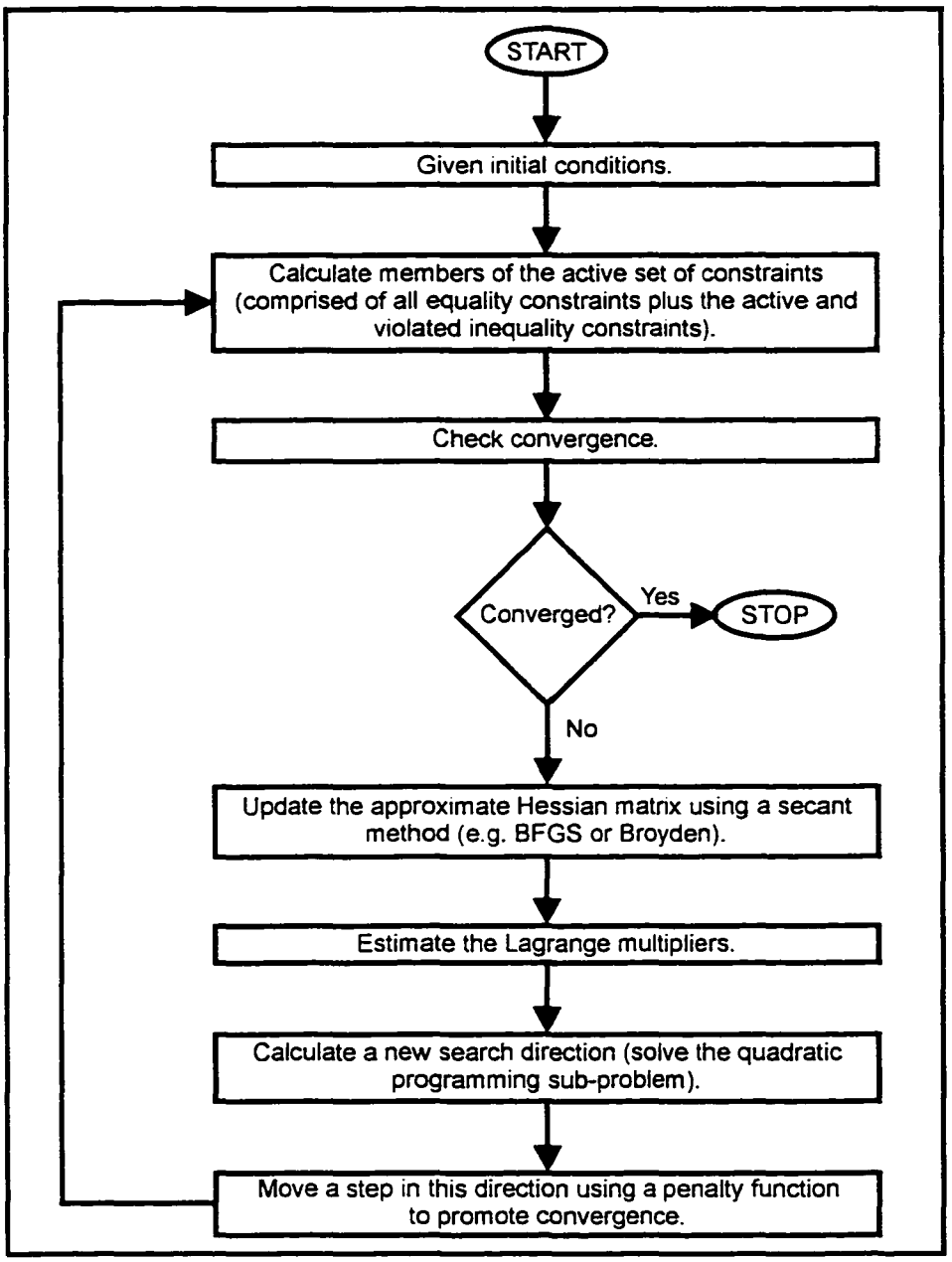


Figure 4.3: Flow diagram summarising a general SQP optimization algorithm [37].

of the objective function. These common blocks need to be declared in both the ASPEN PLUS in-line FORTRAN block and the optimization block.

The reconciled variable values used in the data reconciliation objective function correspond to flowsheet variables in the process model. These ASPEN PLUS flowsheet variables are accessed in the objective function by being declared as FORTRAN variables.

4.2.3 Manipulated Variable Selection

Manipulated variables in ASPEN PLUS must correspond to specified variables in the process model, that is, manipulated variables cannot correspond to calculated (or result) variables. The manipulated variables used in the ASPEN PLUS data reconciliation optimization problem are given in Table 4.1.

The initial guess used for a manipulated variable is the corresponding input specification value. Upper and lower bounds are also required for manipulated variables. These bounds are specified using FORTRAN expressions. The bounds are set to the initial manipulated variable value plus or minus a range. These ranges are specified in the input data file IDFREE.DAT, and read into a FORTRAN common block before the data reconciliation commences. Since this data reconciliation is an unconstrained optimization, these manipulated variable bounds should not be reached during the data reconciliation optimization. The bounds are still required, however, as they are used by the optimization algorithm to help determine manipulated variable scaling and step sizes. Therefore the manipulated variables bounds must be wide enough not restrict the movement of the optimization algorithm, yet narrow enough to prevent too large a step being taken.

4.3 Equation-oriented Simulation

Development of the equation-oriented data reconciliation simulation was performed by editing an ASCII text keyword based SPEEDUP input file. Details of the optimization algorithm used, objective function and manipulated

Table 4.1: ASPEN PLUS data reconciliation manipulated variables.

Variable	**Units**
STAB feed stream mole flow	lbmol/hr
STAB feed stream temperature	°F
C1 molar reflux ratio	lbmol/lbmol
C1 condenser molar vapour fraction	lbmol/lbmol
C1 reboiler mole flow	lbmol/hr
C1 reboiler outlet temperature	°F
E1 outlet temperature	°F
C2 liquid distillate mole flow	lbmol/hr
C2 reboiler mole flow	lbmol/hr
C2 reboiler outlet temperature	°F
C2 condenser sub-cooled temperature	°F
C4 reboiler heat duty	MMBTU/hr
C5 reboiler heat duty	MMBTU/hr
C4 feed stream mole flow	lbmol/hr
C5 feed stream mole flow	lbmol/hr

variable selection are covered in the following subsections.

4.3.1 Optimization Algorithm

The optimization algorithm used to converge the data reconciliation objective function was the SRQP (successive reduced quadratic programming) method [13, 25]. This method is used to converge the process flowsheet in addition to optimizing the data reconciliation objective function. SRQP is an infeasible path optimization method. This means that the optimizer does not necessarily converge the flowsheet, constraints or variable bounds at the end of each iteration. SRQP is particularly suited to the solution of sparse systems with large numbers of equality constraints, typically found in equation-oriented process flowsheets. The steps used by the SRQP optimization algorithm are summarized in Figure 4.4.

The solution of the optimization problem is performed to an optimization tolerance of 1×10^{-1}, that is, until the improvement of the objective function on successive iterations is reduced below this tolerance. The convergence of the flowsheet is performed to a residual tolerance of 1×10^{-5}, that is, until all the flowsheet equation residuals are less than this tolerance. The global convergence scheme chosen for the SRQP method was a line search method with a modified Lagrangian. All other SRQP options were set to their default values.

4.3.2 Objective Function

The objective function in SPEEDUP is specified in the global section of the input file using the same equation syntax used in SPEEDUP models. The objective function itself is the sum of the 28 least square error terms, E_i, corresponding to each measured variable as given in Equation (4.1). These error terms are calculated as additional equality constraints in the Global section. An example error term calculation for the stabilizer feed temperature, T01, is given below:

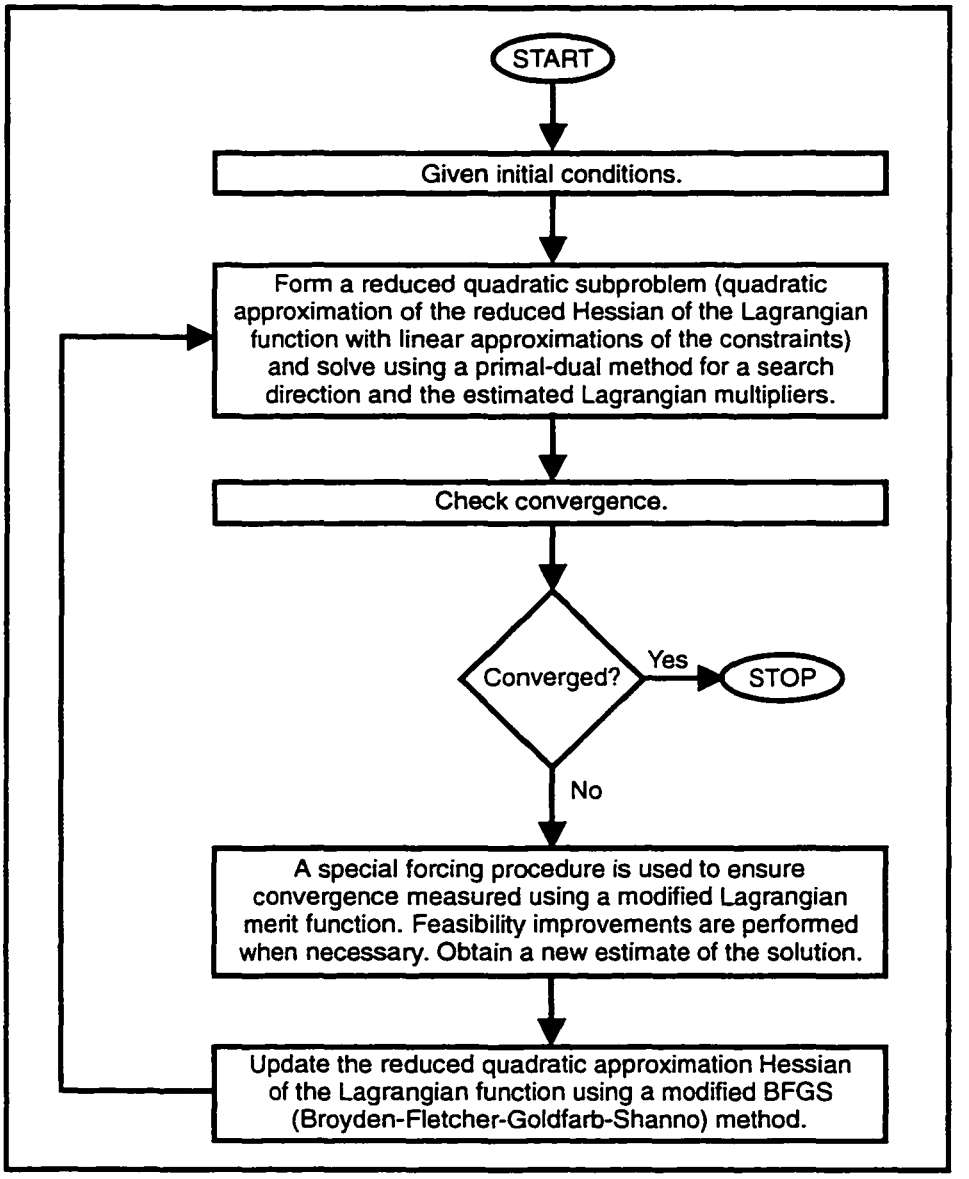


Figure 4.4: Flow diagram summarising the SRQP optimization algorithm [25].

```
ET01 = WT01 * ((MFEED.T_OUT - ST01) / VT01)'2 ;
```

where ET01 = least squares error term corresponding to T01;
WT01 = weighting factor associated with T01;
MFEED.T_OUT = reconciled value of T01;
ST01 = measured value of T01;
VT01 = the standard deviation associated with T01.

The weighting factors, measured values and standard deviations are read from the data files IWGHT.DAT, IMEAS.DAT and IVARNC.DAT respectively using SPEEDUP's external data interface (EDI) (see Appendix C for a detailed description of these data files). The EDI is used prior to the data reconciliation to assign variables with the values read in from external data files. The EDI is specified in two parts. The first is the external section of the SPEEDUP input file which specifies which SPEEDUP variables are to be read in or written out. The second is an EDI FORTRAN input file which is used to interface with the external data, in this case to read from the data files.

The reconciled variable values used in the data reconciliation objective function correspond to flowsheet variables in the process model. These flowsheet variables are accessed directly by using their full SPEEDUP variable name.

4.3.3 Manipulated Variable Selection

Manipulated variables, or "free" variables, must correspond to variables already set in the operation section of the input file. The manipulated variables used in the SPEEDUP data reconciliation optimization problem are given in Table 4.2.

The initial guess used for a manipulated variable is the corresponding value set in the operation section for that variable. Upper and lower bounds are for manipulated variables are specified by adding bound specifications in the set subsection of the operation section. These bound specifications are

Table 4.2: SPEEDUP data reconciliation manipulated variables.

Variable	**Units**
C1 condenser heat duty	MMBTU/hr
DIST-C4 stream mole flow	lbmol/hr
STAB stream mole flow	lbmol/hr
STAB stream temperature	°F
E1 inlet mole flow	lbmol/hr
C1 reboiler mole flow	lbmol/hr
C2 condenser heat duty	MMBTU/hr
C2 condenser subcooled temperature drop (bubble point less subcooled temperature)	°F
E1 outlet temperature	°F
SSFEED stream mole flow	lbmol/hr
C2 reboiler mole flow	lbmol/hr
C4 feed stream mole flow	lbmol/hr
C5 feed stream mole flow	lbmol/hr
HV-ISO stream mole flow	lbmol/hr
LT-DIST stream mole flow	lbmol/hr

set using the EDI before the data reconciliation commences. The bounds are equal to the initial guess for the manipulated variable plus or minus a range. These ranges are specified in the input data file IDFREE.DAT, and are read in and added to or subtracted from the initial guesses using the EDI FORTRAN code. Since this data reconciliation is unconstrained, these manipulated variable bounds should not be reached during the data reconciliation optimization. The bounds are still required, however, as they are used by the optimization algorithm to help determine manipulated variable scaling and step sizes.

4.4 Discussion

The following subsections discuss various problems encountered using both the sequential modular and equation-oriented simulators for data reconciliation. A summary comparing various quantitative aspects of the data reconciliation development and performance are presented in Table 4.3.

4.4.1 Sequential Modular Simulation

The building and debugging of the data reconciliation objective function in ASPEN PLUS was relatively easy. Two problems arose when trying to debug the transfer of data between simulation blocks using the FORTRAN common blocks. The first was caused by the differences between input and result variables in ASPEN PLUS. Input variables are used to specify the problem, and provide initial estimates. However, they are not updated with their final values at the end of a simulation; the final values are stored in a result variable instead. The only exceptions are the optimization manipulated variables which have their final values stored in the input variable. It is very easy to confuse the two types of variables when developing the simulation, thereby passing incorrect data between blocks. Secondly, there is no simple way to check the intermediate values of the data being transferred between blocks. The easiest way to check these variables was to temporarily add FORTRAN

Table 4.3: Comparison of both process simulators for data reconciliation.

	Sequential Modular Simulator	**Equation-oriented Simulator**
Development Time (hours)	133.5	395.5
Input file size (bytes)	63,514	275,288
(without preset variables)	(60,714)	(61,114)
Number of manipulated variables	15	15
Number of reconciled variables	28	28
CPU time (mins)[a]	386.52	106.21
Optimization iterations	15	28
Flowsheet passes or function evaluations	424	50
(tear stream iterations)	(5,660)	
(derivative evaluations)		(29)
Initial objective function value	255.7225	255.72
Final objective function value	41.1884	55.20

[a]CPU time is measured for calculations only on an IBM RS 6000 Model 530H computer.

code which writes variable values to a file to check that the correct values were being passed between blocks. This was time consuming method of checking the data.

Since the variables to be manipulated during the optimization must be chosen from input (or set) variables, the choice of manipulated variables is severely restricted. This is because only certain combinations of variables can be used to specify a given process model in a sequential modular simulation. In addition, even though a given combination of variables are allowed to be specified, convergence of that process model may still be impossible. Initially, the condenser duties for both columns C1 and C2 and the bottoms molar flow rates were used as manipulated variables. Unfortunately convergence of both columns C1 and C2 were not possible using this combination of input variables, even during steady state process model only simulations.

The manipulated variable bounds were initially chosen by guessing how much each variable would be varied during the data reconciliation. These guesses were then adjusted until eventually the variable bounds were wide enough to be inactive throughout the optimization. It was found that if these bounds were too wide, the optimizer would take a large step on the first iteration which sometimes caused the optimization algorithm or flowsheet convergence to fail, thereby terminating the simulation.

The following tuning parameters were used to promote convergence of the SQP optimization algorithm:

- `EST-STEP=YES`. An estimation step is performed before the first iteration to determine the initial step lengths to be taken. This helped prevent large steps being taken on the first iteration of the optimization.

- `DERIV-SWITCH=YES`. When convergence failure occurs, the numerical derivative method is changed from forward difference to central difference. This helped produce better numerical derivative information for the optimizer to use, thereby promoting convergence.

The speed of the optimizer was affected mostly by the tear stream convergence on the splitter column. Since this convergence block was required to converge for each flowsheet evaluation, this resulted in a total of 5,660 passes through the process units C2C4C5, C2SPL1 and F2. Therefore, an average of 13 tear stream iterations were required for each flowsheet pass. To try and reduce the number of tear stream iterations, the SQP optimization algorithm was also used as an infeasible path method. That is, the algorithm was used to solve the tear streams as well, but not necessarily on each flowsheet pass. An extra optimization algorithm tuning parameter, `MAXLSPASS`, was used to specify the maximum number of tear stream iterations allowed on each flowsheet pass. Unfortunately the number required was the same as using SQP as a feasible path method to achieve convergence on the final iteration. Therefore, it was decided use SQP as a feasible path optimizer as there was no performance penalty and tear stream convergence was improved.

4.4.2 Equation-oriented Simulation

Building and debugging of the data reconciliation objective function in SPEEDUP was relatively easy. Transferring data to and from the optimization block (global section) is very simple and easy to check. Variables from other flowsheet blocks are specified directly by typing their full name in the form `unitname.variablename`.

Extreme difficulty was encountered in choosing the manipulated (or free) variables for this flowsheet. Equation-oriented simulators allow any combination of variables to specify the flowsheet, provided they match the degrees of freedom available. However, this does not mean that the flowsheet can necessarily be converged using these variables. SPEEDUP performs checks to see that each equation block is not under or over specified, but does not check for the independence of the manipulated variables. Initially, the same manipulated variables used in the ASPEN PLUS data reconciliation were chosen for the SPEEDUP simulation, however, the flowsheet failed to converge. The manipulated variables used to specify the condensers and reboilers

for C1 and C2 were then changed to achieve flowsheet convergence. However, the optimizer was unable to find a solution and failed for several different reasons including inconsistent constraints, slow convergence and property errors. Slight changes in presets or tuning parameters would cause completely different errors. Eventually it was discovered after analysis of the mass and energy balances around individual process units, that the manipulated variables chosen were completely specifying the mass balance around the stabilizer column and the energy balance around the splitter column. SPEEDUP cannot, however, detect this problem and tries to solve this simulation thereby causing the inconsistent error messages observed. With even larger process flowsheets, this problem could become extremely difficult to detect.

Several other problems were also observed while trying to promote convergence of the data reconciliation problem. The optimization algorithm, SRQP, provides very poor information on its progress when solving. Several print levels are provided which give either too little information, or more information than can reasonably be handled. This makes it very hard to determine if the optimizer has hit variables bounds other than those on the manipulated variables. Also, the SPEEDUP/PROPERTIES PLUS interface is unable to track at which point in the flowsheet that a property errors occur. An error message such as "error in flash" is produced with no indication of which of the 63 flashes in this flowsheet caused the problem. Finally, looking at intermediate values for an infeasible path optimizer solving an equation-oriented simulation flowsheet provides very little useful information. This is because the optimization algorithm does not converge the flowsheet at each iteration, therefore individual variables may have values outside their bounds and may not fulfil mass and energy balances when the optimizer fails. It is sometimes useful to look at equation residuals at the end of an iteration to see which equations are contributing the most error to the solution of the flowsheet. These residuals may indicate problems with either that equation or one of the variables used in that equation.

The manipulated variable bounds were initially chosen by guessing how much each variable would be varied during the data reconciliation. These guesses were then adjusted until eventually they were wide enough to be inactive throughout the optimization. The manipulated variables bounds were found to be critical in preventing the optimization algorithm from taking extremely large steps on the first few iterations. Taking large optimization steps with this process flowsheet tends to cause extremely large flowsheet convergence residual errors which may prevent convergence of the optimizer. In extreme cases, physical property errors occur which generally causes a severe error crashing the SPEEDUP simulation.

Several optimization algorithms were available in SPEEDUP. The standard optimizer, FEASOPT, was unable to even initialize the optimization problem within 8 hours of CPU time. SRQP can be used with three slightly different convergence strategies. The default method uses a line search method with an absolute penalty function to promote convergence. This method tended to take extremely conservative steps and consistently failed due to "slow convergence". The other two methods both use a modified Lagrangian function as the global convergence scheme. The augmented modified Lagrangian function was not required for the data reconciliation optimization problem since no additional inequality constraints were used in the problem.

Tuning of the SRQP optimization algorithm was difficult due to lack of tuning parameters available to modify the behaviour of the optimizer. The only numerical tuning parameters available are the optimization tolerance and the flowsheet residual convergence tolerance. There was no parameter to control the number of iterations taken before the optimization error "slow convergence" is given. There was also no way to directly restrict the manipulated variable step sizes taken to prevent the algorithm from occasionally taking large steps. Scaling of both manipulated variables and process flowsheet variables is not available with SRQP. These would both help convergence of both the flowsheet and optimizer. It is expected that these will

be added in a future release of SPEEDUP.

Some tuning of the optimization problem was possible by varying the data reconciliation weighting factors. The weighting factors were still all equal, but were varied together between 1 and 2 to scale the whole objective function. By varying these weights, it was possible to get convergence for nearly every problem attempted. However, the weights required for convergence were often different for different starting points, and there was no automatic method for choosing these weighting factors. Therefore, since these weights need to be determined manually by the user for each problem attempted, this SPEEDUP simulation is not practically suited for use in a real time optimization system.

4.4.3 Checking for Model Differences

Table 4.4 shows the differences between the ASPEN PLUS and SPEEDUP process models. The results in this table were produced by performing a steady state simulation in ASPEN PLUS using the results obtained at the end of the SPEEDUP data reconciliation. The largest discrepancy found between the two process models for any stream property was a 0.42% difference for the standard volume flow of the Heavy Iso stream. For this operating point, SPEEDUP gave a data reconciliation objective function value of 55.20. ASPEN PLUS calculated the data reconciliation objective function to be 56.04 for the same operating point, a discrepancy of 1.52%. Therefore it can be concluded that the differences between the two process models are extremely small.

Unfortunately ASPEN PLUS does not include any options to view the reduced gradients from the process model during the optimization. This would allow the comparison of the gradients produced by both ASPEN PLUS and SPEEDUP to see if the difference in results were caused by the optimization algorithm or inaccuracies in derivatives from the process models.

Table 4.4: Discrepancies between ASPEN PLUS and SPEEDUP process models at the SPEEDUP data reconciliation optimum.

Stream Name	**Molar Flow**	**Temperature**
Vapour	0.01%	0.00%
C_4-distillate	0.01%	0.00%
Splitter Feed	0.00%	0.00%
Light Iso	0.01%	0.00%
Heavy Iso	0.03%	0.01%
Light Distillate	0.01%	0.00%
SS Feed	0.00%	0.01%

Chapter 5

Economic Optimization Development

5.1 Introduction

5.1.1 Objectives

The objective of the economic optimization is to optimize the process set points to maximize the profitability of the process subject to operating constraints. The adjustments made to the process set points from one cycle of the real time optimizer should be restricted to prevent large changes to the process operating conditions being passed to the plant's process control system. The economic optimization is to be performed using the process model developed in Chapter 3. Figure 5.1 highlights the part of the real time optimizer to be developed in this chapter.

5.1.2 Economic Objective Function

The economic optimization of the stabilizer-splitter process is a nonlinear, steady state, constrained optimization problem. The profit based objective function consists of the sum of the product values less the feed and utility

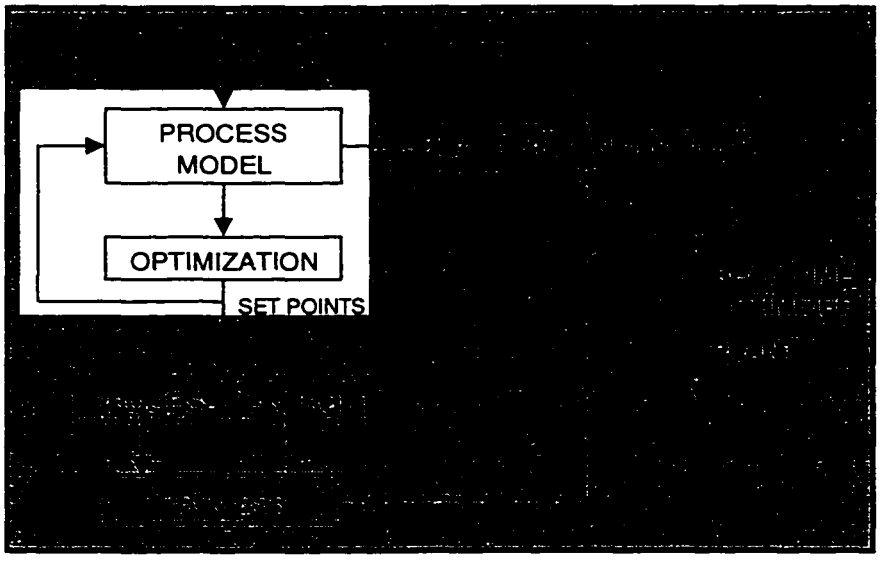


Figure 5.1: Real time optimizer process flow diagram with the economic optimization highlighted.

costs as shown in Equation (5.1).

$$\begin{aligned} &\max_{\mathbf{x}} \left[\sum_{i=1}^{6} P_i - C_{\text{feed}} - C_{\text{utilities}} \right] \\ &\text{subject to } g(\mathbf{x}) \geq \mathbf{0} \\ &\qquad\qquad h(\mathbf{x}) = \mathbf{0} \end{aligned} \tag{5.1}$$

where P_i = product stream values (\$/hr);
C_{feed} = feed stream cost (\$/hr);
$C_{\text{utilities}}$ = total utility costs (\$/hr);
$g(\mathbf{x})$ = the set of process inequality constraints;
$h(\mathbf{x})$ = the set of equality constraints represented by the process model.

The six product values are calculated by the following equation:

$$P_i = D_i v_i \quad i = 1, \ldots, 6 \tag{5.2}$$

where D_i = the product streams unit prices ($/m^3); for the base case, these unit prices are given in Table 5.1 except for the Light Iso product stream which is more complex and is given by Equation (5.3);

v_i = the standard volume flow rate of the product stream (m^3/hr).

The unit cost for the Light Iso product stream also includes a quality premium based on the Reid vapour pressure of the stream. The following equation shows how this quality premium is added to the Light Iso unit price, d_L, from Table 5.1:

$$D_L = d_L + q_L(\mathcal{P}_L - \mathcal{P}_{\text{spec}}) \tag{5.3}$$

where D_L = Light Iso product stream unit price including the Reid vapour pressure quality premium ($/m^3);

q_L = Reid vapour pressure quality premium ($/m^3/kPa);

$\mathcal{P}_L$ = Light Iso product stream Reid vapour pressure (kPa);

$\mathcal{P}_{\text{spec}}$ = Reid vapour pressure specification (kPa).

For the base case, the quality premium, q_L, used was -0.18 $/m^3/kPa, and the Reid vapour pressure specification, $\mathcal{P}_{\text{spec}}$, was 97.0 kPa.

The feed cost is calculated similarly to the product values using the following equation:

$$C_{\text{feed}} = D_{\text{feed}} v_{\text{feed}} \tag{5.4}$$

where D_{feed} = the stabilizer feed stream unit price ($/m^3); for the base case, this unit price is given in Table 5.1;

v_{feed} = the standard volume flow rate of the stabilizer feed stream (m^3/hr).

Finally, the utility cost of the furnace heating oil for the reboilers on both columns C1 and C2 is calculated from Equation (5.5). This method uses the API liquid fuel equivalent (LFE) volume estimated from the reboiler heat duty to determine the utility costs [1].

$$C_{\text{utilities}} = \sum_{i=1}^{2} \left[D_{FO} c_a c_b \left(1/\eta_i\right) Q_i \right] \tag{5.5}$$

Table 5.1: Stream and utility unit prices.

Name	Price	Units
Vapour product stream	0.15	$/m^3
C_4-Distillate product stream	100.07	$/m^3
Light Iso product stream	182.69	$/m^3
Heavy Iso product stream	193.89	$/m^3
Light Distillate product stream	180.52	$/m^3
C2 bottoms product stream	158.34	$/m^3
Stabilizer feed stream cost	158.34	$/m^3
Furnace heating oil cost [a]	82.14	$/LFEm^3

[a]unit price is given in dollars per Liquid Fuel Equivalent standard volume flow

where D_{FO} = the furnace heating oil unit price ($/LFEm^3); for the base case, this unit price is given in Table 5.1;

c_a = volume conversion from barrels to cubic metres (0.1589873 m^3/bbl);

c_b = conversion from heat duty to liquid fuel equivalent volume (0.158730158 MMBTU/LFEbbl);

η_i = furnace efficiency for columns C1 and C2 respectively (assumed to be 70%);

Q_i = furnace heat duty for columns C1 and C2 respectively (MMBTU/hr).

5.1.3 Process Inequality Constraints

In addition to the equality constraints represented by the process model, 25 process inequality constraints were required. These inequality constraints prevent the economic optimization from moving outside plant safety limits, product quality specifications or other operating limits. The inequality constraints required for the stabilizer-splitter process flowsheet are summarised

in Table 5.2.

5.1.4 Set Point Movement Restriction

Bounds restricting the movement of the manipulated variables during the economic optimization are used to prevent large movements in the process set points being returned to the plant after a cycle of the real time optimizer. Large movements in the process set points may upset the plant, affecting the true plant profits or destabilizing the plant's operating conditions. Therefore restriction of the manipulated variable movements is required.

The manipulated variable bounds were chosen from plant operation experience on how large a change can be made to each set point in the plant. For the stabilizer-splitter economic optimization, the manipulated variables were allowed to move by up to $\pm 5.0\%$ of their original values, except temperatures which may move by up to $\pm 5.0^{\circ}$F from their original values during one cycle of the real time optimizer. Therefore, for all manipulated variables except temperatures, the size of the "window" created by the bounds can vary between cycles of the real time optimizer (i.e., $0.05x_0 \neq 0.05x_1$ as shown in Figure 5.2). Figure 5.2 illustrates how these manipulated variable bounds are implemented using an example optimization problem with 2 manipulated variables.

5.2 Sequential Modular Simulation

The sequential modular economic optimization simulation was developed using the MODELMANAGER graphical user interface [8, 9] to produce an ASPEN PLUS input file. Some direct editing of the input file was also required, particularly in developing the in-line FORTRAN blocks, and the FORTRAN objective function.

The SQP optimization algorithm was used to solve this optimization problem. The implementation of the SQP method was similar to that used for the data reconciliation problem; see Section 4.2.1 for more details. Details of

Table 5.2: Economic optimization inequality constraints.

Lower Bound	Variable	Upper Bound	Units
0.005	C_4-Distillate stream C_5 mole fraction	0.150	lbmol/lbmol
—	Light Iso stream Reid vapour pressure	14.0	psi
110.0	Light Iso stream bubble point temperature	—	°F
176.0	Heavy Iso 10% volume distilled true boiling point temperature	266.0	°F
284.0	Heavy Iso 90% volume distilled true boiling point temperature	464.0	°F
302.0	Light Distillate 10% volume distilled true boiling point temperature	554.0	°F
302.0	Light Distillate 90% volume distilled true boiling point temperature	554.0	°F
1000	C1 column reflux standard liquid volume flow	9000	bbl/day
—	C2 column reflux standard liquid volume flow	25,000	bbl/day
—	C4 column feed standard liquid volume flow	10,604	bbl/day
—	C5 column feed standard liquid volume flow	6000	bbl/day
125.0	C1 column tray 3 temperature	220.0	°F
530.0	C1 column reboiler outlet temperature	675.0	°F
120.0	C2 column tray 2 temperature	250.0	°F
550.0	C2 column reboiler outlet temperature	700.0	°F

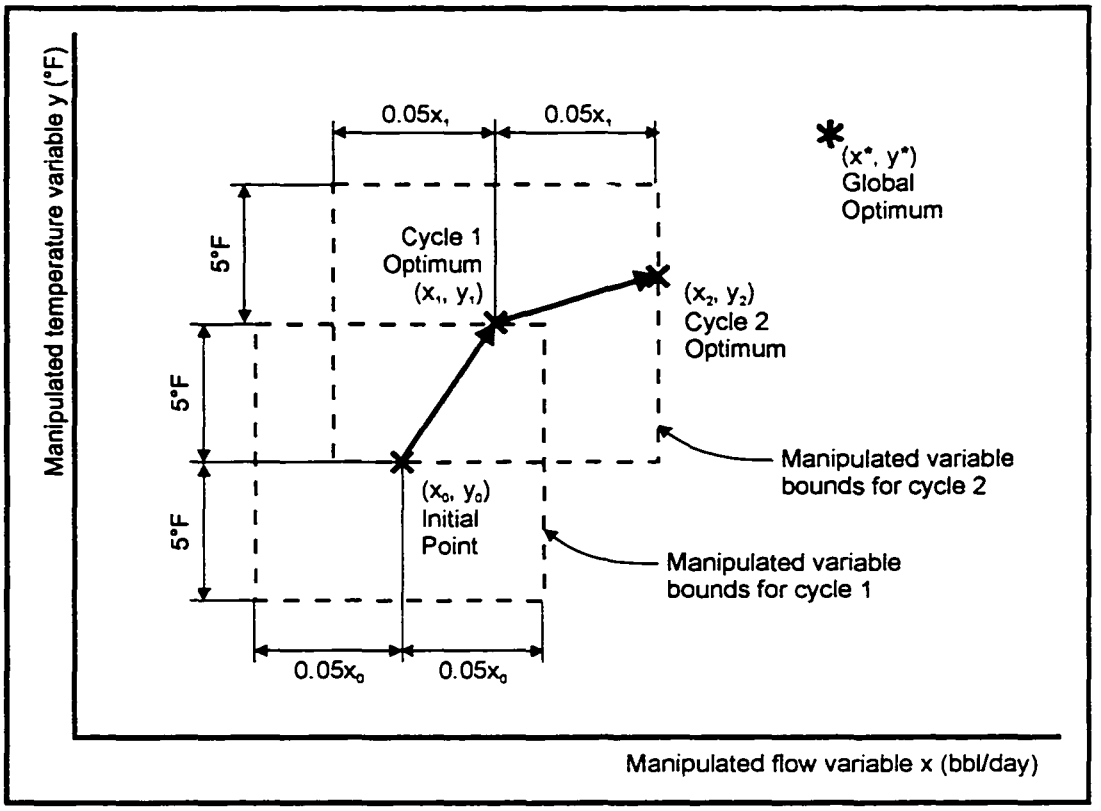


Figure 5.2: Example optimization illustrating the use of manipulated variable bounds. This graph shows how the manipulated variables on cycle 1 of the optimizer are restricted to the dashed box around (x_0, y_0). Cycle 2 of the optimizer starts from the optimum of cycle 1 and is restricted to the second dashed box around (x_1, y_1).

the objective function, inequality constraints and manipulated variables are covered in the following subsections.

5.2.1 Objective Function

The profit objective function in ASPEN PLUS is supplied as FORTRAN 77 code. The objective function is calculated in the following order:

1. Calculate the Light Iso unit price including the Reid vapour pressure quality premium from Equation (5.3);
2. Calculate the six product values individually using Equation (5.2), then sum them to give the total product value;
3. Calculate the feed cost from Equation (5.4);
4. Calculate the two furnace utility costs individually using Equation (5.5);
5. Finally, subtract the feed cost and the two furnace utility costs from the total product value to give the profit to be maximized.

Where necessary, unit conversions were added to the above equations to convert variables from the imperial units used in the process flowsheet to the metric units used in the objective function. The current value of the stream flow rate and reboiler heat duty variables used in the above equations were accessed directly from the process flowsheet. These ASPEN PLUS flowsheet variables are defined as FORTRAN variables prior to execution of the of the objective function code.

The Reid vapour pressure premium (q_L) and specification ($\mathcal{P}_{\text{spec}}$), unit prices given in Table 5.1 and the furnace efficiencies (η_i) are all read in from the data file IOBJFN.DAT using an in-line FORTRAN block (see Appendix B for a detailed description of this data file). This in-line FORTRAN block is executed before the economic optimization commences. The data read in from the file are stored in FORTRAN "common" blocks for use during the calculation of the objective function.

5.2.2 Process Inequality Constraints

The 25 inequality constraints listed in Table 5.2 were implemented using 25 constraint blocks in ASPEN PLUS. Each constraint block is also listed in the optimization block in order to apply the constraint to that optimization problem. Each constraint requires a specification involving ASPEN PLUS flowsheet variables defined as FORTRAN variables, and FORTRAN common block variables. A tolerance specification for each constraint is also required. The constraint is said to be "active" if the constrained variables are within this tolerance, "inactive" if the variables are inside the constraint and "violated" if the variables are outside the constraint.

The constraint values given in Table 5.2 are read in from the data file ICNSTR.DAT using an in-line FORTRAN block (see Appendix B for a detailed description of this data file). The data read in from the file are stored in FORTRAN "common" blocks for use during the calculation of the constraints.

5.2.3 Manipulated Variable Selection and Bounding

The manipulated variables used in the ASPEN PLUS economic optimization are the same as those used in the data reconciliation, except the stabilizer feed stream flow rate and temperature. The stabilizer feed stream conditions cannot be affected by the stabilizer-splitter process, and therefore cannot be manipulated during the economic optimization. The manipulated variables used are summarised in Table 5.3. The initial guesses used for the manipulated variables are the corresponding process flowsheet input specification values.

The condenser heat duties for columns C1 and C2 cannot be varied automatically by the process control system in this plant. Unfortunately, these heat duties are result variables in the process model and therefore cannot be fixed easily. Therefore, two equality constraint blocks were added to prevent movement of the condenser heat duties during the economic optimization.

Table 5.3: ASPEN PLUS economic optimization manipulated variables.

Variable	Units
C1 molar reflux ratio	lbmol/lbmol
C1 condenser molar vapour fraction	lbmol/lbmol
C1 reboiler mole flow	lbmol/hr
C1 reboiler outlet temperature	°F
E1 outlet temperature	°F
C2 liquid distillate mole flow	lbmol/hr
C2 reboiler mole flow	lbmol/hr
C2 reboiler outlet temperature	°F
C2 condenser sub-cooled temperature	°F
C4 reboiler heat duty	MMBTU/hr
C5 reboiler heat duty	MMBTU/hr
C4 feed stream mole flow	lbmol/hr
C5 feed stream mole flow	lbmol/hr

These constraint blocks work by storing the initial value of the condenser heat duty to a FORTRAN common block variable before the optimization commences. The constraint is then specified by requiring the condenser heat duty to be equal to its initial value stored in the common block variable. to within the specified tolerance of 0.1 MMBTU/hr.

The upper and lower bounds on the manipulated variables are used to restrict the movement of the optimizer to prevent large changes in the set points being returned to the plant's process control system (see Section 5.1.4 for more detail). These manipulated bounds are specified for all manipulated variables except temperatures similarly to the following example ASPEN PLUS input code:

```
LIMITS "C2MD-(C2MD*PC2MD)" "C2MD+(C2MD*PC2MD)"
```

where `C2MD` = initial manipulated variable value (for this example, the Light Iso stream molar flow rate);

PC2MD = the fraction of the initial manipulated value value that this variable my be varied. For this problem, PC2MD is equal to 0.05 for all manipulated variables except temperatures.

For temperature manipulated variables, the manipulated variable bounds are specified in the same manner as the following ASPEN PLUS input code:

```
LIMITS "C1RBT - PC1RBT" "C1RBT + PC1RBT"
```

where C1RBT = initial manipulated variable value (for this example, the C1 reboiler outlet temperature);
PC1RBT = the number of degrees Fahrenheit this manipulated variable may be varied. For this problem, PC1RBT is equal to 5°F for all manipulated temperature variables.

The variables specifying the ranges in which the manipulated variables may move (for the above example codes, PC2MD and PC1RBT) are read in from the data file IOFREE.DAT using an in-line FORTRAN block (see Appendix B for a detailed description of this data file). The data read in from this file are stored in FORTRAN "common" blocks for use during the calculation of the manipulated variable bounds.

5.3 Equation-oriented Simulation

Development of the equation oriented economic optimization was performed by editing the ASCII text keyword based SPEEDUP input file. Details of the objective function, inequality constraints and manipulated variables are covered in the following subsections.

5.3.1 Optimization Algorithm

The SRQP optimization algorithm was used to solve the SPEEDUP economic optimization problem. The implementation of the SRQP method was similar

to that used for the data reconciliation problem, except that the global convergence scheme chosen was a line search method with a modified augmented Lagrangian. The modified augmented Lagrangian was used because of the inequality constraints present in this optimization problem See Section 4.3.1 for more details on how the SRQP algorithm was implemented.

5.3.2 Objective Function

The objective function was specified in the global section of the SPEEDUP input file using the same equation syntax used in SPEEDUP models. The objective function to be maximized is the profit calculated by subtracting the feed cost and the two furnace utility costs from the total product value. This profit objective is also multiplied by a weighting factor which can be varied to help promote convergence (see Section 5.4.2 for more detail). The following equations are included as equality constraints in the global section of the input file to be solved simultaneously with the rest of the process flowsheet:

- Calculate the Light Iso unit price including the Reid vapour pressure quality premium from Equation (5.3);
- Calculate the six product values individually using Equation (5.2), then sum them to give the total product value;
- Calculate the feed cost from Equation (5.4);
- Calculate the two furnace utility costs individually using Equation (5.5);

Where necessary, unit conversions were added to the above equations to convert variables from the imperial units used in the process models to the metric units used in the objective function. The current value of the stream flow rate and reboiler heat duty variables used in the above equations were accessed directly from the process flowsheet by using their full SPEEDUP variable name.

The Reid vapour pressure premium (q_L) and specification ($\mathcal{P}_{\text{spec}}$), unit prices given in Table 5.1, furnace efficiencies (η_i) and the objective function weighting factor are all read in from the data file IOBJFN.DAT using SPEEDUP's external data interface (EDI) (see Appendix C for a detailed description of this data file). The EDI is used prior to the optimization to assign SPEEDUP variables with variables read in from external data files.

5.3.3 Process Inequality Constraints

The 25 inequality constraints listed in Table 5.2 were implemented in the global constraint subsection of the SPEEDUP input file. Each inequality constraint is also specified as a variable bound for the corresponding constrained variable through the EDI. This method of duplicating constraints in both the global section and as a variable bounds is recommended when using SRQP. This is because SRQP can handle simple variable bounds directly without the use of slack variables [13]. These simple bounds will always be satisfied by the SRQP method, unlike the inequality constraints.

The constraint values given in Table 5.2 are read in from the data file ICNSTR.DAT using SPEEDUP's EDI (see Appendix C for a detailed description of this data file).

5.3.4 Manipulated Variable Selection and Bounding

The manipulated variables used in the SPEEDUP economic optimization are the same as those used in the data reconciliation, except the stabilizer feed stream flow rate and temperature, and the condenser heat duties. The stabilizer feed stream conditions cannot be affected by the stabilizer-splitter process, and therefore cannot be manipulated during the economic optimization. Likewise, the condenser heat duties for the columns C1 and C2 cannot be adjusted automatically by the process control system in this plant and are therefore not manipulated during the economic optimization. The manipulated variables used are summarised in Table 5.4. The initial guesses

Table 5.4: SPEEDUP economic optimization manipulated variables.

Variable	Units
DIST-C4 stream mole flow	lbmol/hr
E1 inlet mole flow	lbmol/hr
C1 reboiler mole flow	lbmol/hr
C2 condenser subcooled temperature drop (bubble point less subcooled temperature)	°F
E1 outlet temperature	°F
SSFEED stream mole flow	lbmol/hr
C2 reboiler mole flow	lbmol/hr
C4 feed stream mole flow	lbmol/hr
C5 feed stream mole flow	lbmol/hr
HV-ISO stream mole flow	lbmol/hr
LT-DIST stream mole flow	lbmol/hr

used for the manipulated variables are the corresponding input specification values.

The upper and lower bounds on the manipulated, or free, variables are used to restrict the movement of the optimizer to prevent large set point changes as detailed in Section 5.1.4. From testing of this optimization problem in SPEEDUP, it was found that the SRQP optimization algorithm was unable to find an optimum if it lay on a manipulated variable bound. Therefore the manipulated variable bounds were applied indirectly as bounds and inequality constraints on flowsheet variables equal to the manipulated variables. For example, the column C1 bottoms mole flow, `C1BT29.L_OUT1` is a manipulated variable. The heat exchanger E1 inlet mole flow, `E1.F_IN` should also be at the same value as the bottoms mole flow, therefore both bounds and global inequality constraints are placed on this variable instead. Similar equivalent variables were found for all of the manipulated variables in this optimization.

The upper and lower bounds, and inequality constraints on the variables equivalent to the manipulated variables were implemented in the optimization simulation as follows:

- For all variable types except temperature, the upper and lower bounds and inequality constraints are calculated from:

$$\begin{aligned} y_i^{UB} &= x_i^{(0)} + q_i x_i^{(0)} \\ y_i^{LB} &= x_i^{(0)} - q_i x_i^{(0)} \end{aligned} \tag{5.6}$$

  where y_i^{UB} = upper bound for the variable equivalent to the manipulated variable;
  y_i^{LB} = lower bound for the variable equivalent to the manipulated variable;
  $x_i^{(0)}$ = initial manipulated variable value:
  q_i = the fraction of the initial manipulated variable value that this variable may be varied. For this problem, all q_i are equal to 0.05.

- For temperature variables, the upper and lower bounds and inequality constraints are calculated from:

$$\begin{aligned} y_i^{UB} &= x_i^{(0)} + r_i \\ y_i^{LB} &= x_i^{(0)} - r_i \end{aligned} \tag{5.7}$$

  where r_i = the number of degrees Fahrenheit this manipulated variable may be varied. For this problem, all r_i are equal to 5°F.

The variables specifying the ranges in which the manipulated variables may move (the values q_i and r_i in Equations (5.6) and (5.7)) are read in from the data file IOFREE.DAT using SPEEDUP's EDI (see Appendix C for a detailed description of this data file). The values of the bounds y_i^{UB} and y_i^{LB} are calculated in the EDI FORTRAN code and then assigned directly to the variable's upper and lower bounds, and also to SPEEDUP global variables for use in the inequality constraints.

The manipulated variables also require upper and lower bounds for use by the optimization algorithm to determine initial step sizes. These bounds, however, should not be reached during the optimization simulation. Therefore, these bounds are set outside the bounds placed on the variables equivalent to the manipulated variables. This is achieved by the following method:

- For all variable types except temperature, the upper and lower manipulated variable bounds are calculated from:

$$\begin{aligned} x_i^{UB} &= x_i^{(0)} + 1.5\, q_i x_i^{(0)} \\ x_i^{LB} &= x_i^{(0)} - 1.5\, q_i x_i^{(0)} \end{aligned} \tag{5.8}$$

  where x_i^{UB} = upper bound for the manipulated variable:
  x_i^{LB} = lower bound for the manipulated variable.

- For temperature variables, the upper and lower bounds and inequality constraints are calculated from:

$$\begin{aligned} x_i^{UB} &= x_i^{(0)} + 1.5\, r_i \\ x_i^{LB} &= x_i^{(0)} - 1.5\, r_i \end{aligned} \tag{5.9}$$

The values of the bounds x_i^{UB} and x_i^{LB} are also calculated in the EDI FORTRAN code and then assigned directly to the manipulated variable's upper and lower bounds.

5.4 Discussion

The following subsections discuss various problems encountered using both the sequential modular and equation-oriented simulators for economic optimization. A summary comparing various quantitative aspects of the economic optimization development and performance are presented in Table 5.5.

Table 5.5: Comparison of both process simulators for economic optimization.

	Sequential Modular Simulator	**Equation-oriented Simulator**
Development Time (hours)	73.5	134.5
Input file size (bytes)	40,746	291,130
(without preset variables)	(37,616)	(71,374)
Number of manipulated variables	13	11
Number of inequality constraints	25	47
Number of additional equality constraints	2	0
CPU time (mins)[a]	119.78	104.89
Optimization iterations	6	28
Flowsheet passes or function evaluations	139	55
(tear stream iterations)	(1,519)	
(derivative evaluations)		(29)
Initial objective function value ($/hr)	-159.7225	-159.73
Final objective function value ($/hr)	91.1353	416.00
Number of constraints active at solution	1[b]	1[c]

[a]CPU time is measured for calculations only on an IBM RS 6000 Model 530H computer.
[b]Light Iso stream Reid vapour pressure upper bound constraint active.
[c]C_4-Distillate stream C_5 component mole fraction lower bound constraint active.

5.4.1 Sequential Modular Simulation

The building and debugging of the economic optimization simulation was relatively simple. A problem arose when trying to find a method of preventing the column C1 and C2 condenser heat duties from being varied during the economic optimization. Ideally, the condenser heat duties would have been specified variables in process flowsheet preventing them from changing throughout the simulation. However, flowsheet convergence was unable to be obtained when the condenser heat duties were used as specified variables. Therefore, two equality constraints were added to ensure that the condenser heat duties remained at their initial value. It was found that unless this constraint was 'converged' on the first iteration of the optimizer, it was extremely unlikely that the SQP optimization algorithm would be able to satisfy the constraint before the end of the simulation. The solution to this was to ensure that the initial values of the condenser heat duties were passed to the optimization simulation from the results of the previous data reconciliation simulation. This ensured that both of the equality constraints were converged on the first iteration of the SQP algorithm, and usually remained converged throughout the optimization.

The only other problems with the ASPEN PLUS economic optimization were found when trying to debug errors in passing information between simulation blocks using FORTRAN common blocks. This was the same problem encountered with the data reconciliation simulation development, and was solved in the same way using temporary FORTRAN code to print out the values of the variables being passed between simulation blocks (see Section 4.4.1 for more detail).

One extra SQP optimization algorithm tuning parameter was used in addition to those described in Section 4.4.1 for the data reconciliation simulation. This parameter was `CONST-ITER` which was set to 10 iterations. This parameter specifies the number of additional iterations that the SQP optimization algorithm should take after the optimization convergence test has been satisfied to satisfy any violated constraints. This parameter was

increased to 10 iterations to help improve the optimizer's ability to converge any violated constraints

Most convergence problems encountered when testing this optimization simulation were caused by violated constraints. It was found that if the simulation began with significantly violated inequality constraints that the SQP optimization algorithm appeared to have some difficulty in satisfying these constraints during the course of the optimization. The errors returned by the SQP optimization algorithm were either "Up hill search direction predicted" or "Line search failed on the final iteration". This was especially noticeable on violations of the constraints involving "property set" variables: Reid vapour pressure, 10% and 90% volume distilled true boiling point temperatures, and bubble point temperature. Property set variables are measurements defined by the user to be calculated for flowsheet streams other than the standard flow, composition and thermodynamic data produced by ASPEN PLUS. Increasing the parameter `CONST-ITER` helped. but not in all cases.

As found with the data reconciliation optimization in ASPEN PLUS. computational speed of the optimizer was affected mostly by the tear stream convergence on the splitter column. A total of 1519 passes through the process units C2C4C5, C2SPL1 and F2 were required for the 139 flowsheet evaluations required during the optimization. Therefore an average of 11 tear stream iterations were required for each flowsheet pass requiring a significant amount of computational time.

5.4.2 Equation-oriented Simulation

Development of the objective function and inequality constraints was relatively straight forward for the SPEEDUP economic optimization. The manipulated variables were chosen to be the same as the data reconciliation optimization except for the stabilizer feed stream mole flow and temperature, and the column C1 and C2 condenser heat duties. This prevented the convergence problems due to poor choice of manipulated variables as seen in

the data reconciliation optimization development. Also, since the column C1 and C2 condenser heat duties were already specified variables, there was no need for equality constraints to prevent the economic optimizer from varying these variables as with ASPEN PLUS.

The main problem encountered in debugging the SPEEDUP economic optimization was restricting the movement of the manipulated variables. Initially, the manipulated variable bounds were applied directly to the manipulated variables. This resulted in repeatable errors from the SRQP optimization algorithm caused by "inconsistent constraints" when the optimum was found to lie on a manipulated variable bound. Since the manipulated variable bounds are being used deliberately to limit the movement of the economic optimization on each cycle of the real time optimizer, several manipulated variable bounds will often be active at the optimum. To alleviate this problem, indirect manipulated variable bounds were added to prevent the true manipulated variable bounds from being reached as described in Section 5.3.4. These indirect bounds were implemented as both bounds and global inequality constraints on the variables equivalent to the manipulated variables. The reason for both bounds and inequality constraints is because the variable bounds can be handled directly by the SRQP optimization algorithm, and presence of the inequality constraints provide Lagrangian and other information at the end of the optimization. This is the same reason for duplicating the process inequality constraints in both the global constraint subsection and also as variables bounds. No reason has been found for why the SRQP optimization algorithm consistently fails when the optimum lies on a manipulated variable bound, however the above work around does appear to fix this problem.

Tuning of the SRQP optimization algorithm was again found to be difficult for the same reasons found in the data reconciliation optimization, mostly due to a lack of tuning parameters with which to modify the algorithms behaviour. The global convergence scheme chosen was the augmented modified Lagrangian function due to the presence of inequality constraints.

It was found that the line search method with an absolute penalty function tended to take very small step sizes and consistently failed due to "slow convergence".

The profit objective function was multiplied by a weighting factor to provide a method for tuning the optimization. This weighting factor was varied between 1 and 2 to promote convergence of the economic optimization. Generally the weighting factor was found to lie between 1.2 and 1.5. This method worked for most of the problems attempted, but often required retuning of the weighting factor even for slight changes in starting point. Convergence failures for problems with poorly tuned weighting factors were reported by the SRQP optimization algorithm to be caused by either "slow convergence" or "inconsistent constraints". No consistent pattern or reason for these convergence failures could be found. A possible reason may be the unscaled manipulated and process flowsheet variables in this problem. Also the criteria for "slow convergence" appears to be too strict and often occurs when the optimizer seems close to an answer. Unfortunately, there is no way to tune the SRQP algorithm's criteria for slow convergence to prevent this error.

From Table 5.5, it can be seen that SPEEDUP achieved a significantly better profit objective function value for this one run of the economic optimization. This higher objective function profit may have been caused by a combination of SPEEDUP's infeasible path SRQP optimization method and more accurate derivative information available from the process flowsheet models. See Section 6.6.2 for a more detailed comparison of the ASPEN PLUS and SPEEDUP economic optimization results.

5.4.3 Checking for Model Differences

Table 5.6 shows the differences between the ASPEN PLUS and SPEEDUP process models. The results in this table were produced by performing a steady state simulation in ASPEN PLUS using the results obtained at the end of the SPEEDUP economic optimization. The largest discrepancy found between the two process models for any stream property was a 0.93%

Table 5.6: Discrepancies between ASPEN PLUS and SPEEDUP process models at the SPEEDUP economic optimization optimum.

Stream Name	Molar Flow	Temperature
Vapour	0.00%	0.00%
C_4-distillate	0.00%	0.00%
Splitter Feed	0.01%	0.00%
Light Iso	0.00%	0.00%
Heavy Iso	0.00%	0.00%
Light Distillate	0.02%	0.00%
SS Feed	0.00%	0.00%

difference for the Heavy Iso stream 10% volume distilled true boiling point temperature. For this operating point, SPEEDUP gave a profit objective function value of \$416.00/hr. ASPEN PLUS calculated the profit to be \$411.23/hr for the same operating point, a discrepancy of 1.15%. Therefore it can be concluded that the differences between the two process models are extremely small.

Unfortunately ASPEN PLUS does not include any options to view the reduced gradients from the process model during the optimization. This would allow the comparison of the gradients produced by both ASPEN PLUS and SPEEDUP to see if the difference in results were caused by the optimization algorithm or inaccuracies in derivatives from the process models.

Chapter 6

Real Time Optimizer Linking and Results

6.1 Real Time Optimizer Linking

The data reconciliation and economic optimization simulations developed in the previous two chapters need to be linked together to form the real time optimizer. The linking of the optimizer involves writing a program to call the data reconciliation and economic optimization simulations when required, transfer data between the simulations and the plant, store the results produced and detect any errors encountered during the real time optimization. Figure 6.1 highlights the part of the real time optimizer to be developed in this section.

The data to be transferred within the real time optimization system is similar for both real time optimizers. Figure 6.2 illustrates the data files which need to be transferred between the data reconciliation and economic optimization simulations, and also between the real time optimizer and the plant's process control system. The real time optimizer cycle begins with the data reconciliation module reading in the measurement data from the plant's process control system. For these simulations, the process data is supplied either from process data provided by Shell Canada, or from a steady state

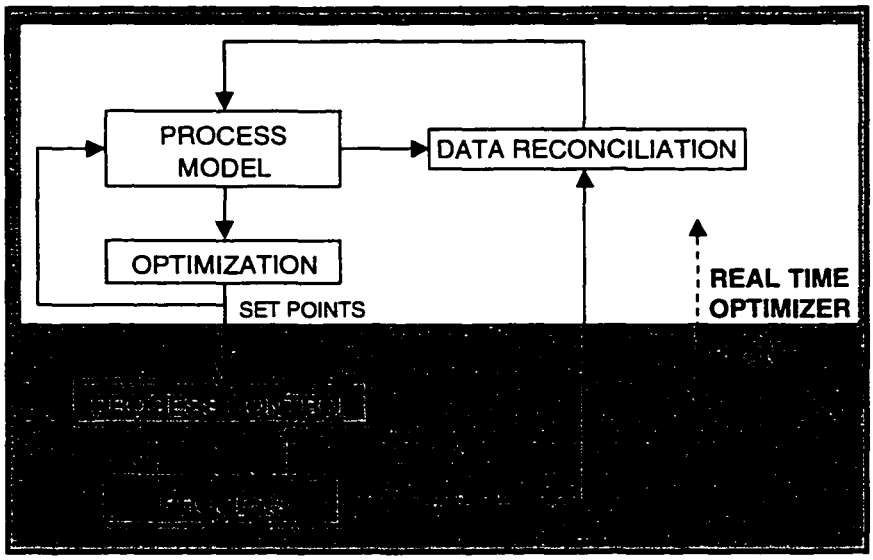


Figure 6.1: Real time optimizer process flow diagram with the real time optimizer highlighted.

process simulation acting as the plant. The Shell process data are averaged over one hour of measurements from the plant and are used for the first cycle of most of the following real time optimizer runs. All subsequent cycles use data provided from the steady state simulation either with or without random noise added (see Section 6.2 for details of the noise added).

The data reconciliation simulation reads in the measurement data variances and data reconciliation weights from data files. Before the data reconciliation commences, the final simulation settings from the previous economic optimization simulation are read in from a file. These simulation settings are used to provide the best known starting point for the data reconciliation. When the data reconciliation is completed, the reconciled variables and other results files are produced by the simulation. The simulation settings at the end point of this simulation are also transmitted to the economic optimization simulation to provide the starting point.

The economic optimization simulation then reads in the economic data

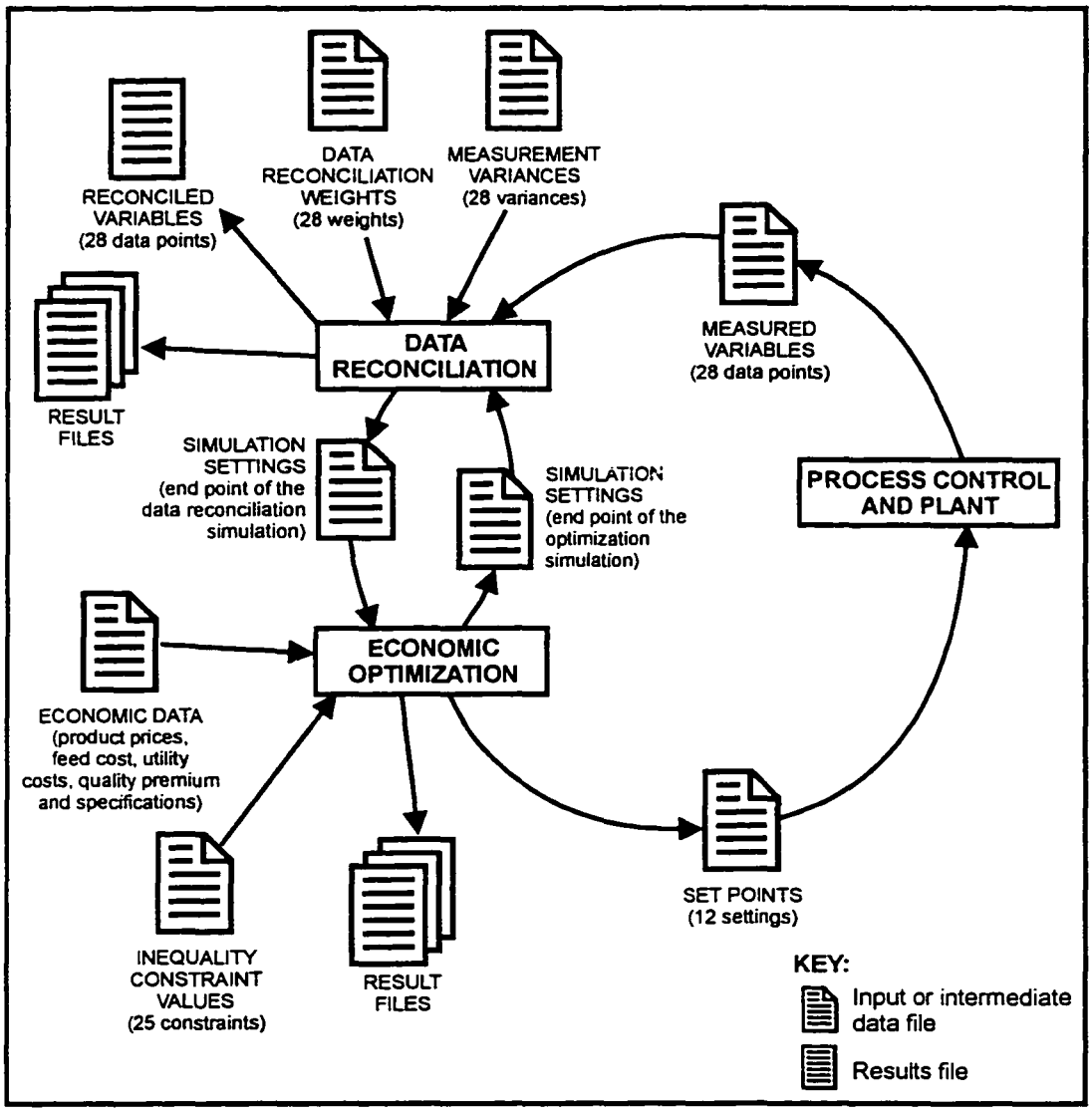


Figure 6.2: Real time optimizer data transfer.

and inequality constraint values from data files. At the end of the optimization, the new optimized set points for the process are sent to the plant's process control system for implementation in the plant. Also, the simulation settings from the end point of the optimization are stored in a data file for use by the data reconciliation simulation during the next cycle of the real time optimizer.

As previously mentioned, most testing of the real time optimizer is performed using measurement data from a steady state ASPEN PLUS simulation of the stabilizer-splitter process. This simulation is used to simulate the set points received from the economic optimization and output a new set of measurement data for the next cycle. Random noise is added to the process data using a small FORTRAN program which calls a NAG FORTRAN Library Routine [84] for generating random numbers. The input code for both the plant simulation and the FORTRAN program to add noise are included in Appendix F.

6.1.1 Sequential Modular Simulator

Figure B.2 in Appendix B shows a flow chart of the sequential modular simulator implementation of the real time optimizer cycle illustrated in Figure 6.2. A Unix C-Shell script is used to control the real time optimizer by starting the simulations when required, moving the data files between simulations, storing the results and checking for errors. The data reconciliation and economic optimization simulations were combined into a single input file and sequenced using an ASPEN PLUS sequence block. This allows the economic optimization to start using the end point of the data reconciliation without the need to write out to a data file. Passing the results of the economic optimization back to the data reconciliation for the next cycle was more difficult. Every specified and estimated variable needs to written out and read in through a user specified FORTRAN block. This involved a lot of code and was difficult to debug.

Before running the real time optimizer, a setup program needs to be ex-

ecuted which translates and compiles the simulations. Appendix B provides the documentation for setting up and running the ASPEN PLUS based real time optimizer. The input code used for the sequential modular real time optimizer is included in Appendix D.

6.1.2 Equation-oriented Simulator

Figure C.2 in Appendix C shows a flow chart of the equation-oriented simulator implementation of the real time optimizer cycle illustrated in Figure 6.2. Similar to the sequential modular simulator, a Unix C-Shell script is used to control the real time optimizer by starting the simulations when required, moving the data files between simulations, storing the results and checking for errors. SPEEDUP has no sequencing feature, so the data reconciliation and economic optimization have to be run as separate simulations in different directories. Therefore the results of the data reconciliation have to written to file and passed on to the economic optimization simulation. However, this was accomplished simply by issuing a single command which saves the results of previous run to file. These results are then read back in and are then used to preset a simulation using SPEEDUP's "Use" environment. To run SPEEDUP simulations in batch mode, a text file storing the commands for the SPEEDUP Executive is specified in the command line when running the simulation.

Before running the real time optimizer, a setup program needs to be executed which translates and compiles the simulations, and runs the PROPERTIES PLUS input files. Appendix C provides the documentation for setting up and running the SPEEDUP based real time optimizer. The input code used for the equation-oriented real time optimizer is included in Appendix E.

6.2 Real Time Optimizer Comparison

The first run comparing the sequential modular and equation-oriented real time optimizers used the base case economic data presented in Table 5.1. The

two optimizers were run for as many cycles as possible using this economic data with no random noise added to the process measurements. That is, the economic optimization on each cycle was started with the results of the previous economic optimization. The term "cycle" of a real time optimizer will be used in this thesis to refer to one complete run through the flow chart given in Figure 6.1. That is, the measured data is read in from the plant, data reconciliation is performed on this data, then the process is optimized before the set points are passed back to the plant.

Figure 6.3 graphs the progress of the economic objective function on each cycle for both of the real time optimizers. The expected global optimum profit of $751.95/hr shown in Figure 6.3 was the highest obtained profit over several runs for this problem when all restrictions on the movement of the manipulated variables were removed (the process inequality constraints were still applied). Since the actual shape of the shape of the objective function is extremely complex, and also because of any process modelling errors, this optimum cannot be guaranteed to be the true global optimum for this process.

SPEEDUP failed after fewer cycles, unable to converge the economic optimization on sixth cycle, however, it achieved a higher profit objective than the ASPEN PLUS real time optimizer, nearly reaching the expected global optimum profit. Twelve different weighting factors were tried manually in the economic objective function, all of which ended with the same "slow convergence" error given by the SRQP optimization algorithm. It is possible that a weighting factor exists which can converge this problem, however it was extremely difficult to determine the value of the weighting value manually for this particular optimization.

ASPEN PLUS failed on the eleventh cycle due to a violated inequality constraint which the SQP algorithm could not satisfy. The violated constraint was the upper limit on the Light Distillate stream 90% volume distilled true boiling point temperature. The error given by the optimization algorithm was "line search failed on final iteration; solution may not be op-

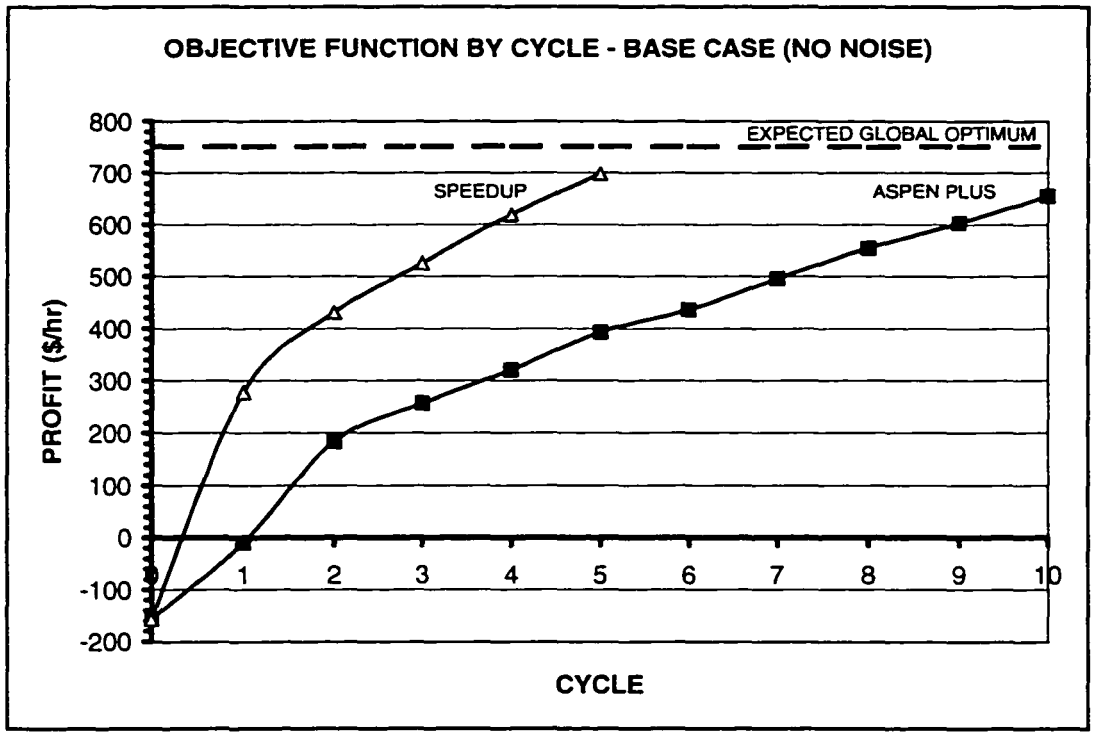


Figure 6.3: Graph comparing base case with no measurement noise profits (economic objective function) for both real time optimizers. The expected global optimum profit for this optimization is shown as a dashed line.

timal".

The active inequality constraint at the end of the first cycle was the Light Iso stream Reid vapour pressure upper bound for both real time optimizers. For all subsequent cycles of the ASPEN PLUS real time optimizer, the active constraint was the upper bound on the Light Distillate stream 90% volume distilled true boiling point temperature. The upper bound constraint on the Light Iso stream Reid vapour pressure was also active at the end of cycles 2, 6, 7 and 10. For all cycles, except the first, of the SPEEDUP real time optimizer, the active inequality constraints were lower bound on the C_4-Distillate stream C_5 mole fraction and the upper bound on the Light Distillate stream 90% volume distilled true boiling point temperature. Figures 6.4 and 6.5 show the set point changes made at the end of each cycle by both of the real time optimizers. For more information on the performance of the two real time optimizers, see Table A.3 and Table A.4 in Appendix A.

A second run comparing the two types of real time optimizer was performed on the base case economic data with noise added to the plant measurements. Again, the two optimizers were run for as many cycles as possible using the economic data presented in Table 5.1. The random noise was added to the process measurements calculated by the plant simulation was normally distributed with a standard deviation equal to 5% of the measured value.

A graph comparing the profit objective function values achieved by the two real time optimizers is given in Figure 6.6. As can be seen from this graph, both optimizers managed to complete only a few real time optimization cycles. Again, the expected global optimum of \$751.95/hr is shown in Figure 6.6.

The SPEEDUP real time optimizer failed on the third cycle due to severe property errors which could not be corrected, however, it still attained a higher profit than the ASPEN PLUS optimizer. The property error occurred during a flash calculation by PROPERTIES PLUS when the bubble point for a mixture could not be found. Unfortunately there is no way to determine which flash in the flowsheet was consistently causing this error during

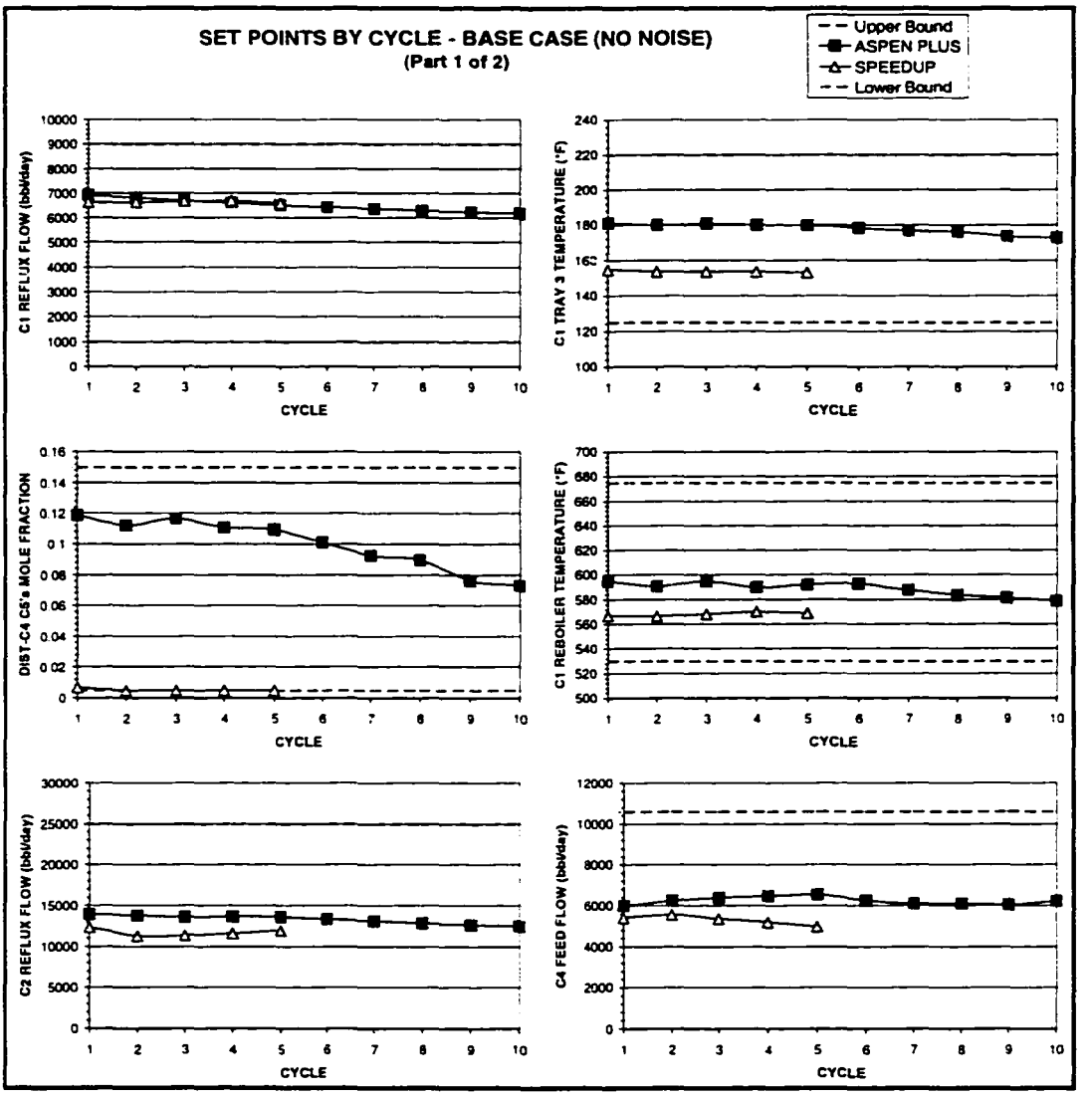


Figure 6.4: Graphs of the set point changes made during the base case runs with no measurement noise (1 of 2).

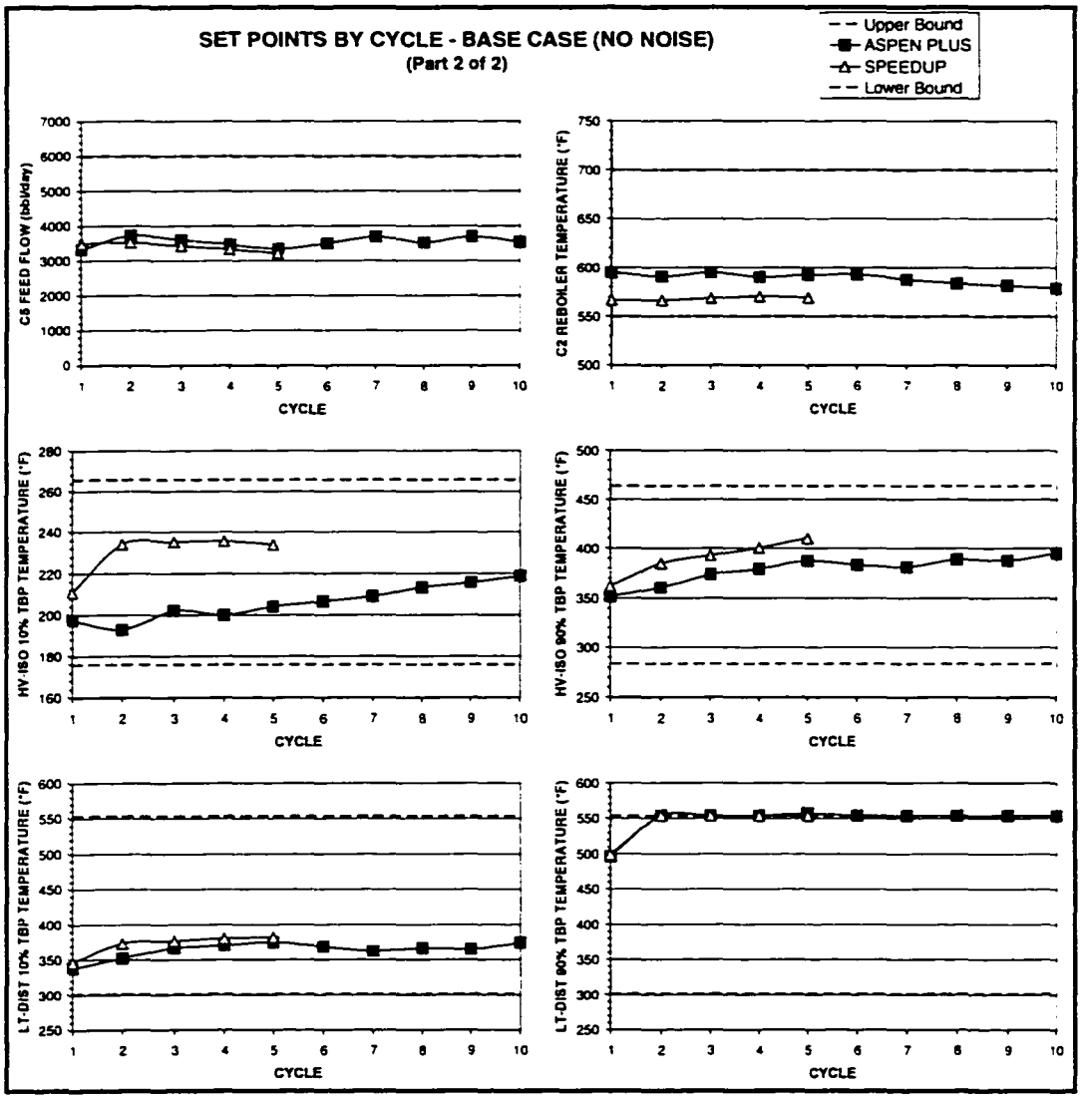


Figure 6.5: Graphs of the set point changes made during the base case runs with no measurement noise (2 of 2).

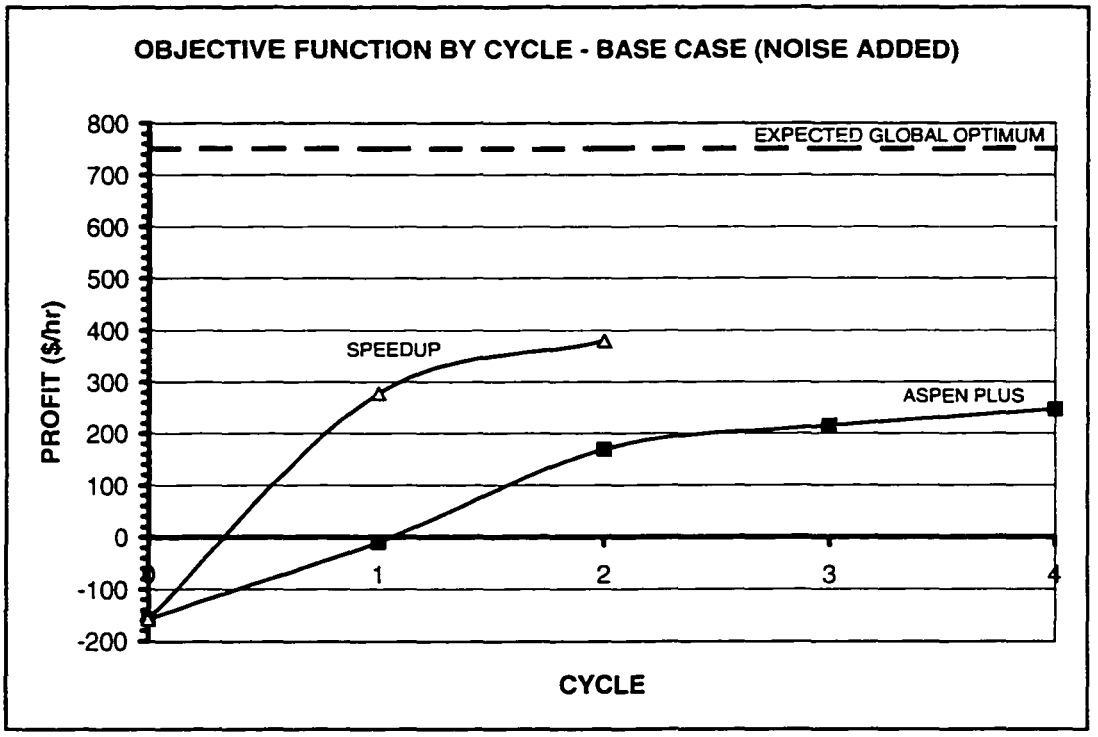


Figure 6.6: Graph comparing base case profits (economic objective function) with measurement noise for both real time optimizers. The expected global optimum profit for this optimization is shown as a dashed line.

the economic optimization on the third cycle of the real time optimizer. It should be noted that SPEEDUP required the objective function weights for both data reconciliation and economic optimization objective functions to be changed on both cycles completed. Changing the economic objective function weight for the third cycle of the real time optimizer had no effect on the property errors generated.

The ASPEN PLUS real time optimizer failed on the fifth cycle due to a violated inequality constraint which the SQP optimization algorithm could not satisfy. The violated bound was the upper limit on the Light Distillate stream 90% volume distilled true boiling point temperature. The error given by the optimization algorithm was "line search failed on final iteration; solution may not be optimal".

The active inequality constraint at the end of the first cycle was the Light Iso stream Reid vapour pressure upper bound for both real time optimizers. For all subsequent cycles of the ASPEN PLUS real time optimizer, the active constraint was the upper bound on the Light Distillate stream 90% volume distilled true boiling point temperature. At the end of the second cycle of the SPEEDUP real time optimizer, the active inequality constraints were lower bound on the C_4-Distillate stream C_5 mole fraction and the upper bound on the Light Distillate stream 90% volume distilled true boiling point temperature. For more information on the performance of the two real time optimizers, see Table A.1 and Table A.2 in Appendix A.

6.3 Economic Model Changes

Three areas of the economic model were studied:

1. Changes to the two key product stream prices, Heavy Iso and Light Distillate.

2. Sensitivity analysis of the economic objective function.

3. The effect of changes in the economic parameters part way through a real time optimizer run. This was tested by changing the product prices from typical winter operating prices to typical summer operating prices. Please note that these runs were tested using ASPEN PLUS version 9.3 instead of version 9.2 as all other runs used. For more details on the different versions of ASPEN PLUS, see Section 6.5.

The results of these studies are presented in the following subsections. Only the ASPEN PLUS real time optimizer was used in the following studies due to the problems encountered tuning the SPEEDUP real time optimizer for each cycle.

6.3.1 Product Price Changes

Four real time optimizer runs were performed to study the effect of Heavy Iso and Light Distillate product stream price changes. High and low values were chosen for each of these product prices to see what effect they have on the operation of the real time optimizer. The high and low prices were \$193.89/m^3 and \$173.89/m^3 for the Heavy Iso product stream and \$200.52/m^3 and \$180.52/m^3 for the Light Distillate product stream respectively. The run with the Heavy Iso and Light Distillate stream prices at \$193.89/m^3 and \$180.52/m^3 respectively corresponds to the base case run in Section 6.2. Random measurement noise was added to the data produced by the "plant" simulation for all four of the following real time optimizer runs.

A graph comparing the objective function profits at the end of each cycle for each of the four real time optimizer runs is presented in Figure 6.7. It should be noted that the negative profits shown in Figure 6.7 mean that the stabilizer-splitter process is being run at a loss, however, this does not necessarily reflect a loss for the refinery as a whole. This is because the feed flow rate to this process is being determined by an upstream process and is not being varied by the real time optimizer. Since the feed to the stabilizer-splitter process cannot be used as a product, it must be split into products

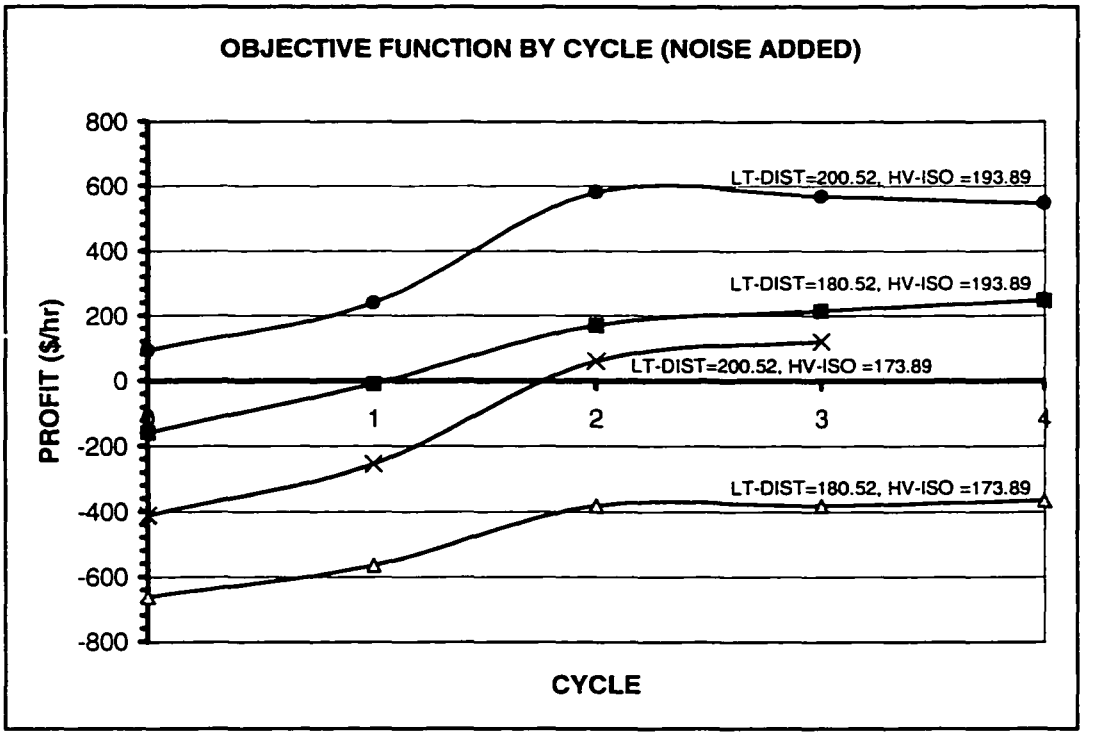


Figure 6.7: Graph comparing profits (economic objective function) for different Heavy Iso and Light Distillate prices using the ASPEN PLUS real time optimizer.

even if that incurs a loss for this process. In this case the real time optimizer is trying to minimize the loss incurred by the stabilizer-splitter process.

The real time optimizer runs shown in Figure 6.7 were attempted for four cycles each, however, the case where the Heavy Iso and Light Distillate prices were $\$173.89/m^3$ and $\$200.52/m^3$ respectively, convergence of the economic optimization on the fourth cycle was not possible. This convergence failure was for the same reason as the previous ASPEN PLUS real time optimizer failures, that is, the SQP optimization algorithm was unable to satisfy the upper bound inequality constraint on the Light Distillate stream 90% volume distilled true boiling point temperature. The error given by the optimization

algorithm was "no feasible point found to quadratic subproblem".

The active inequality constraints at the end of each real time optimizer cycle were the same as the base case run except for the case where both Heavy Iso and Light Distillate stream prices were at their high values. For all cases, the active inequality constraint after the first cycle was the Light Iso stream Reid vapour pressure upper bound, and at the end of the other cycles, the upper bound on the Light Distillate stream 90% volume distilled true boiling point temperature was active. For the case where the Heavy Iso and Light Distillate prices were \$193.89/m^3 and \$200.52/m^3 respectively, at the end of the fourth cycle, the lower bound on the Heavy Iso stream 10% volume distilled true boiling point temperature was also active. Figures 6.8 and 6.9 show the changes in set points made at the end of each cycle of the four real time optimization runs completed. For more information on the performance of real time optimizer on each of the four cases studied, see Tables A.1, A.5, A.6 and A.7 in Appendix A.

6.3.2 Economic Objective Function Sensitivity Analysis

To perform the economic objective function sensitivity analysis, the economic optimization simulation developed in Chapter 5 was modified to allow free movement of the manipulated variables. That is, the bounds used to constrict the manipulated variable movements on each real time optimizer cycle were widened to allow the economic optimization to proceed directly to the optimum subject to the inequality constraints. The product stream prices, feed stream cost and utility fuel oil costs were then varied by $\pm 5\%$ from their original values (the original prices and costs are given in Table 5.1). The results of this study were produced by Fenton [39] and are summarised in Table 6.1. Each of the 14 runs detailed in Table 6.1 began from the simulation settings of an optimized base case run. The initial profit before and after optimization with the new product price or cost is given, along with

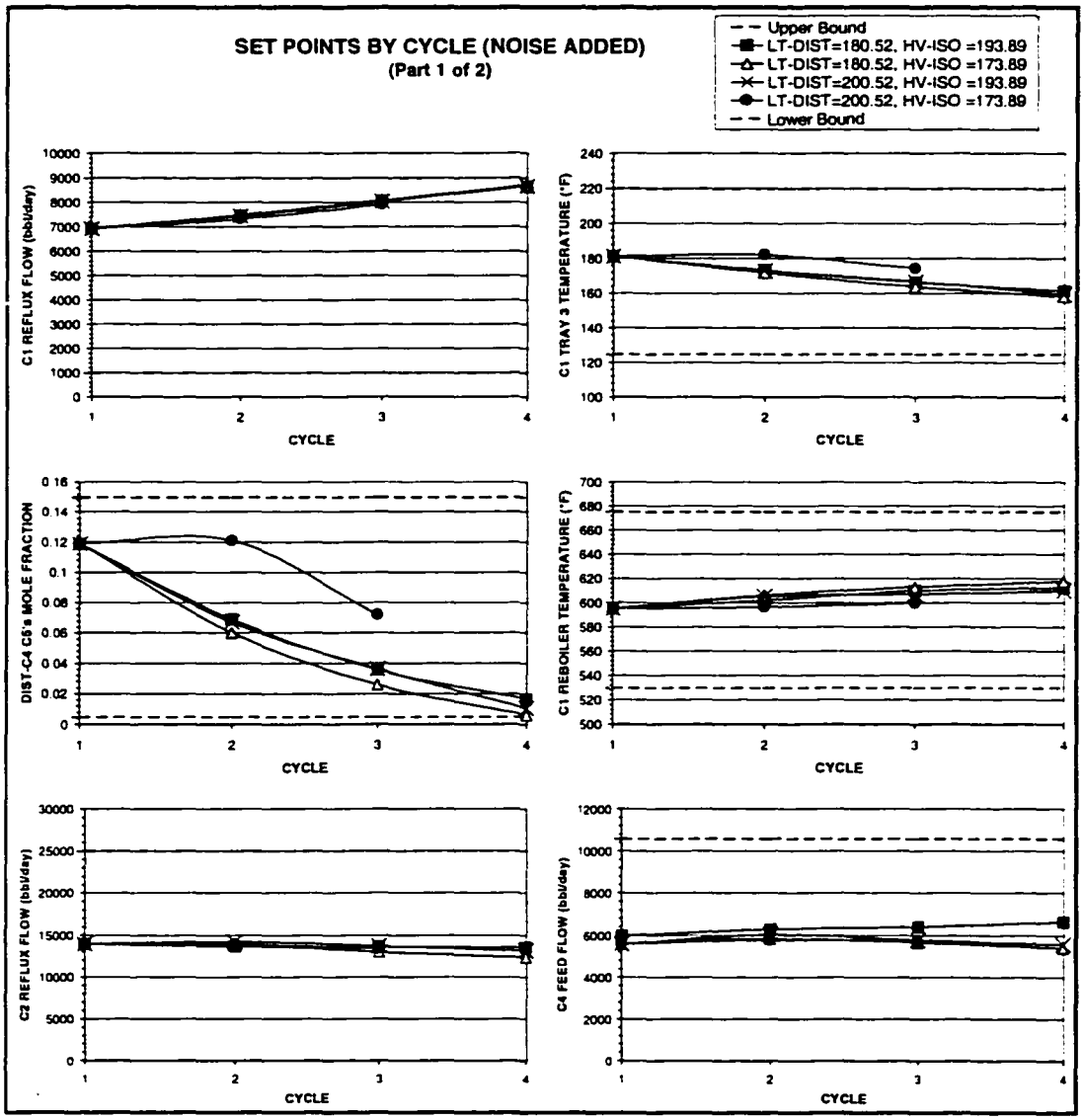


Figure 6.8: Graphs of the set point changes made during the runs with different Heavy Iso and Light Distillate prices using the ASPEN PLUS real time optimizer (1 of 2).

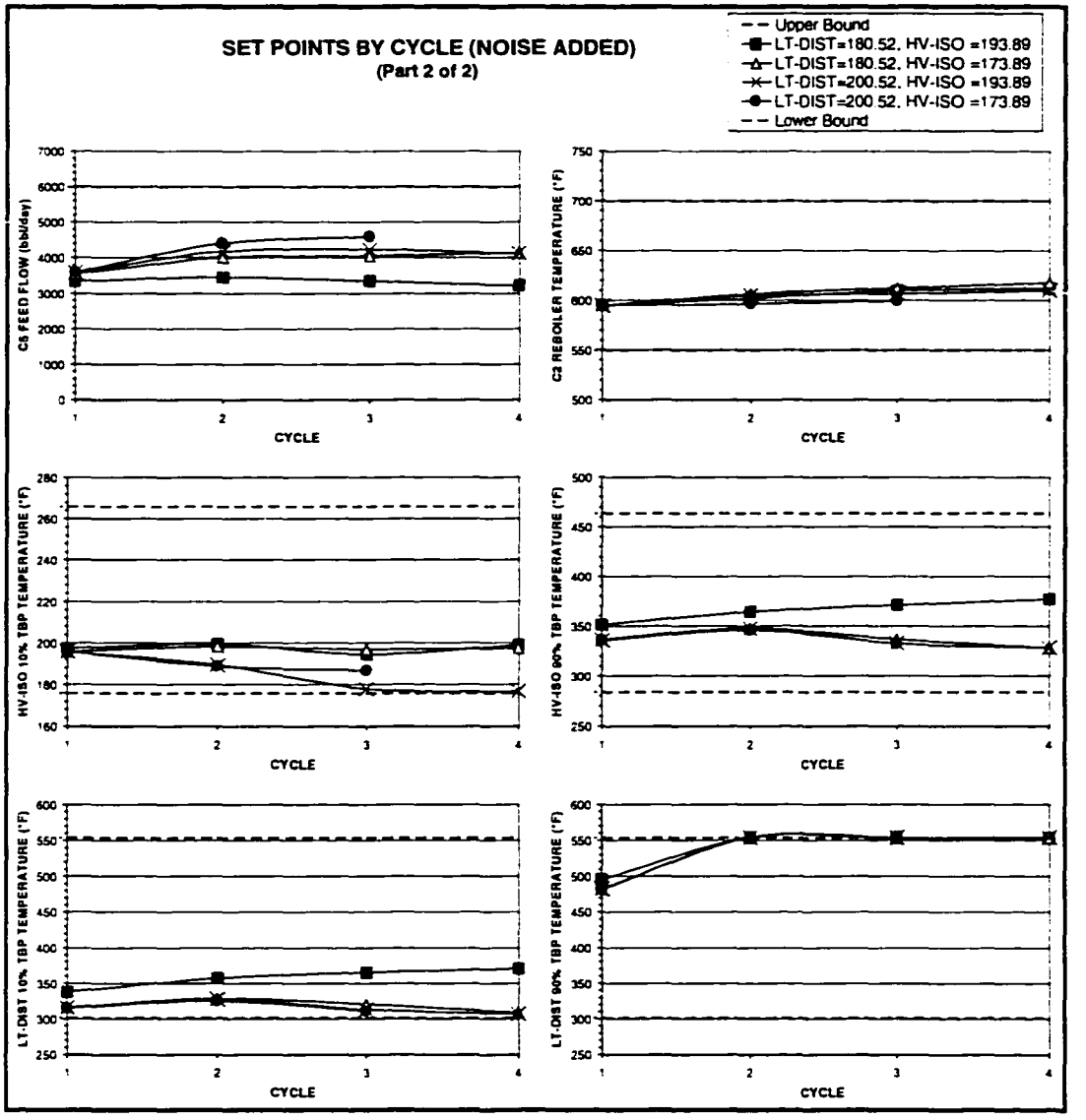


Figure 6.9: Graphs of the set point changes made during the runs with different Heavy Iso and Light Distillate prices using the ASPEN PLUS real time optimizer (2 of 2).

the penalty for not re-optimizing after this price or cost change occurred.

6.3.3 Changes in Economics During Real Time Optimization Runs

Four real time optimizer runs were performed using the ASPEN PLUS version 9.3 real time optimizer instead of version 9.2 (see Section 6.5 for more details on the changes between versions). Two of the runs used winter economic conditions throughout, that is, the prices were \$173.89/m^3 and \$200.52/m^3 for Heavy Iso and Light Distillate streams respectively and a Light Iso product Reid vapour pressure upper limit was set to 14 psi. The other two runs were performed with product prices changing part way through the run to summer prices from the winter conditions given above. The summer conditions used were \$193.89/m^3 and \$180.52/m^3 for Heavy Iso and Light Distillate streams respectively and a Light Iso product Reid vapour pressure upper limit of 10.5 psi. The price changes were chosen to reflect the change between typical summer and winter economic conditions, that is, in winter heating oil is generally in higher demand, while in summer, gasoline is in higher demand.

Graphs comparing the profit objective functions at each cycle for the winter case and winter to summer price change runs, with and without measurement noise, are presented in Figure 6.10. The ASPEN PLUS version 9.3 real time optimizer failed on the fifth cycle of the winter case run with random measurement noise due to two violated constraints. For the winter to summer run with measurement noise, the real time optimizer failed on the sixth cycle, also due to two violated inequality constraints. In both cases, the constraints violated were the upper bound on the stabilizer column C1 reflux flow and the upper bound on the Light Distillate stream 90% volume distilled true boiling point temperature. The error returned by the SQP optimization algorithm for the winter case was "no feasible point found to quadratic subproblem", and for the winter to summer run the error returned

Table 6.1: ASPEN PLUS economic objective function sensitivity study results [39].

Run	**Profit before optimization ($/year)**	**Profit after optimization ($/year)**	**Penalty for not optimizing ($/year)**	**Penalty (%)**
Base case	—	4,901,100	—	—
Low Vapour price	4,901,000	5,530,800	628,700	12.8
High Vapour price	4,901,200	5,137,500	236,300	4.8
Low C_4-Distillate price	4,375,200	4,477,400	102,200	2.3
High C_4-Distillate price	5,427,100	5,574,900	147,800	2.7
Low Light Iso price	4,227,300	4,663,500	436,200	10.3
High Light Iso price	5,575,000	7,548,700	1,973,700	35.4
Low Heavy Iso price	3,025,800	4,004,100	978,300	32.3
High Heavy Iso price	7,450,300	9,717,200	2,266,900	30.4
Low Light Distillate price	4,152,400	4,663,400	511,000	12.3
High Light Distillate price	5,649,900	8,066,800	2,416,900	42.8
Low Stabilizer feed cost	9,110,600	9,269,600	159,000	1.7
High Stabilizer feed cost	691,600	1,832,800	1,141,200	165.0
Low fuel oil cost	4,947,900	7,257,000	2,309,100	46.7
High fuel oil cost	4,854,400	6,380,200	1,525,800	31.4

was "line search failed on final iteration; solution may not be optimal".

The winter case and the winter to summer run with no measurement noise were both completed without errors for all 10 cycles of the real time optimizer attempted. However, the winter to summer with no measurement noise did have violated constraints at the end of both cycles 4 and 5, the two cycles completed after the change to summer conditions. The constraints violated at the end of the fourth cycle were the Light Iso Reid vapour pressure and the Light Distillate stream 90% volume distilled true boiling point temperature upper bounds. At the end of the fifth cycle, just the Light Iso Reid vapour pressure constraint was violated.

The active inequality constraints for most of the winter run with random measurement noise were the C_4-Distillate stream C_5 mole fraction lower bound and the Light Distillate stream 90% volume distilled true boiling point temperature upper bound. For the winter run with no measurement noise, the active inequality constraints were the Light Iso stream Reid Vapour pressure for the first few cycles. For the latter cycles, the active constraints were both the upper and lower bounds on the Light Distillate stream 90% volume distilled true boiling point temperature. For more information on the performance of real time optimizer on the winter case runs, see Tables A.8 and A.9 in Appendix A.

For the winter to summer run with noise added to the measurement data, the active constraints for most of the summer cycles was the upper limits on the Light Distillate stream 90% volume distilled true boiling point temperature and the Light Iso stream Reid vapour pressure. The Light Distillate stream 90% volume distilled true boiling point temperature was at its upper limit for all of the summer cycles of the winter to summer run without measurement noise. For the first 4 summer cycles, the Light Iso stream Reid vapour pressure was also at its upper limit. For more information on the performance of real time optimizer on the two winter to summer runs, see Tables A.10 and A.11 in Appendix A.

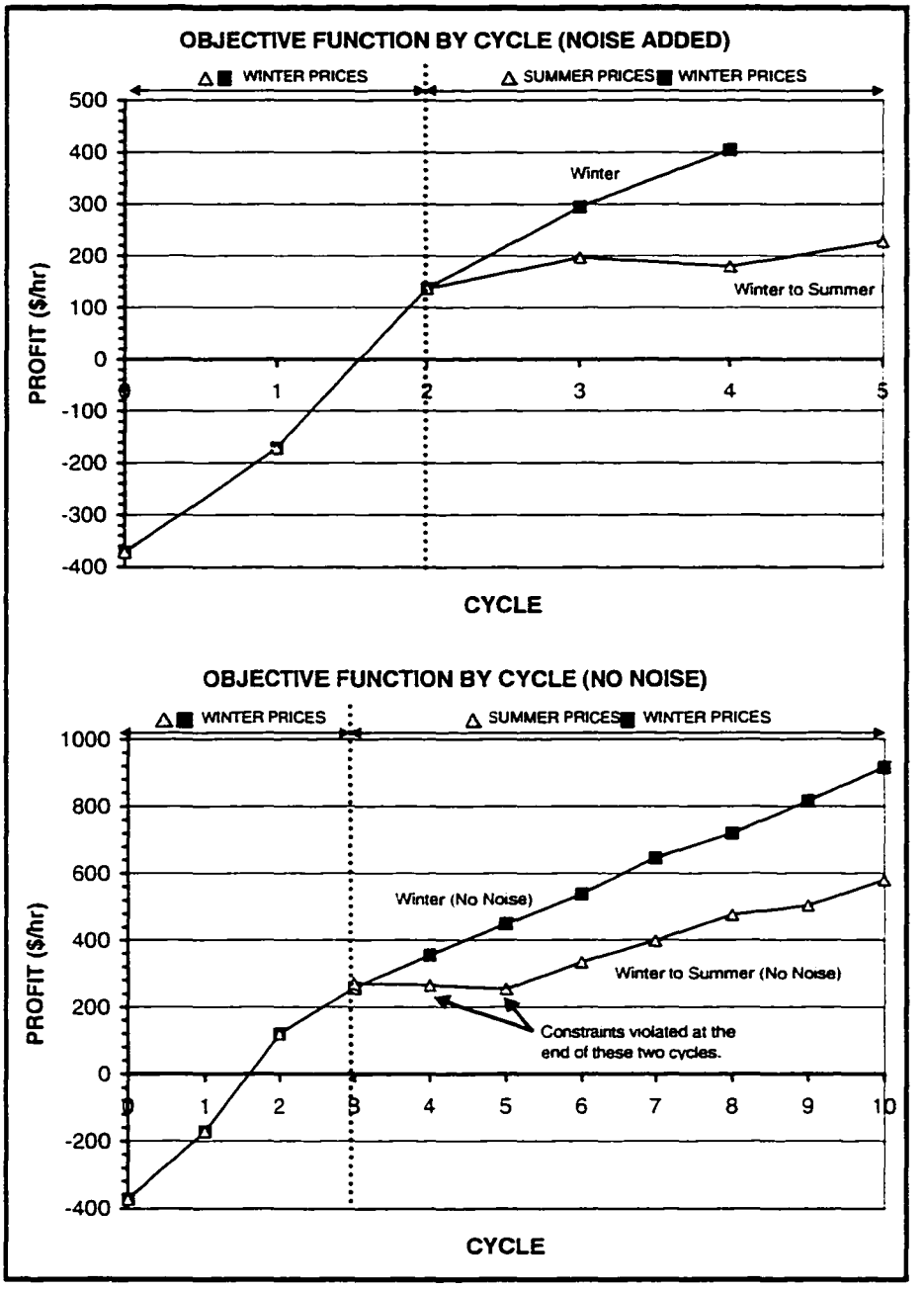


Figure 6.10: Graphs comparing profits (economic objective function) for winter case and winter to summer price change with and without noise using the ASPEN PLUS version 9.3 real time optimizer.

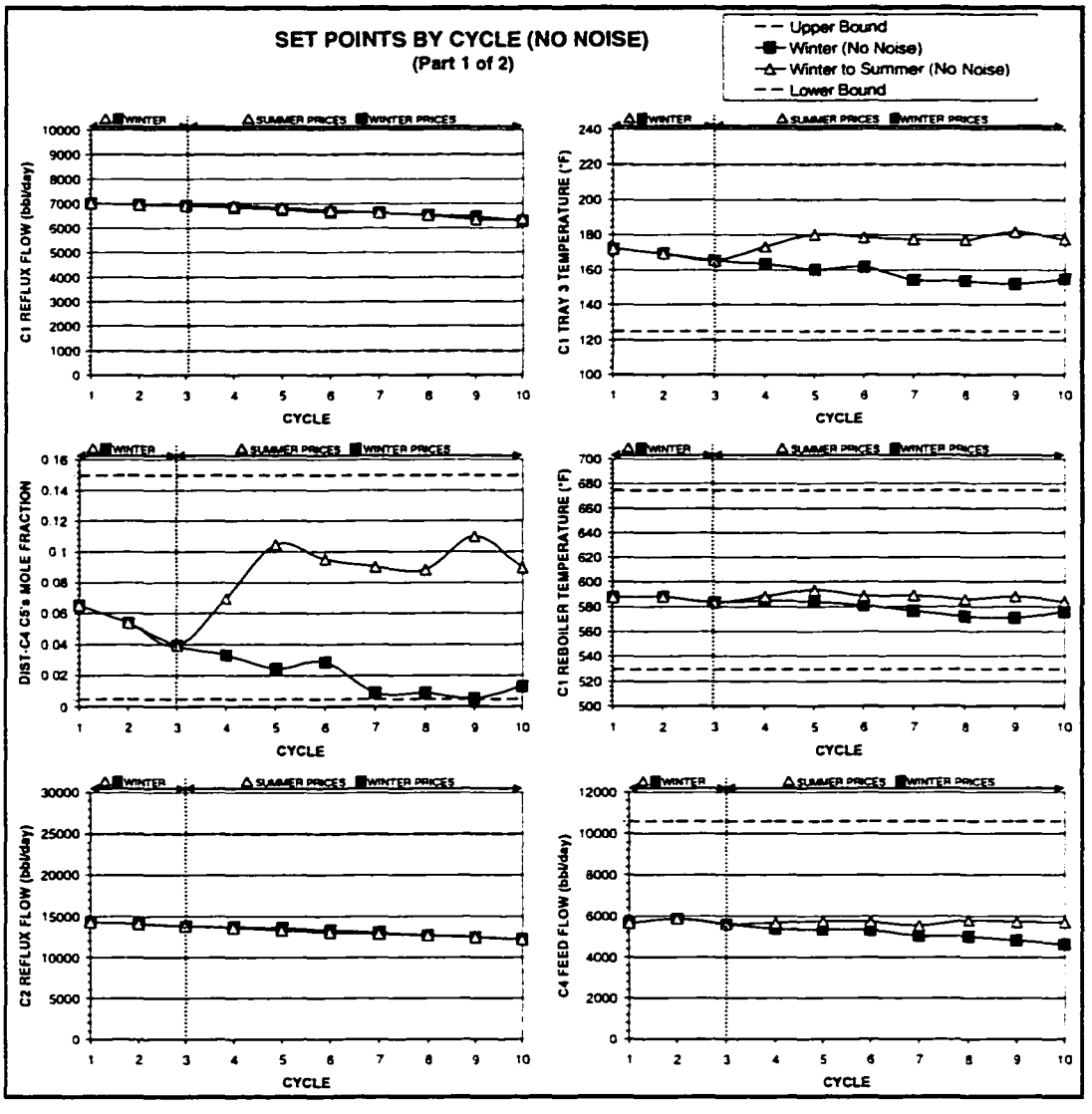


Figure 6.11: Graphs of the set point changes made during the winter case and winter to summer price change with no measurement noise runs using the ASPEN PLUS version 9.3 real time optimizer (1 of 2).

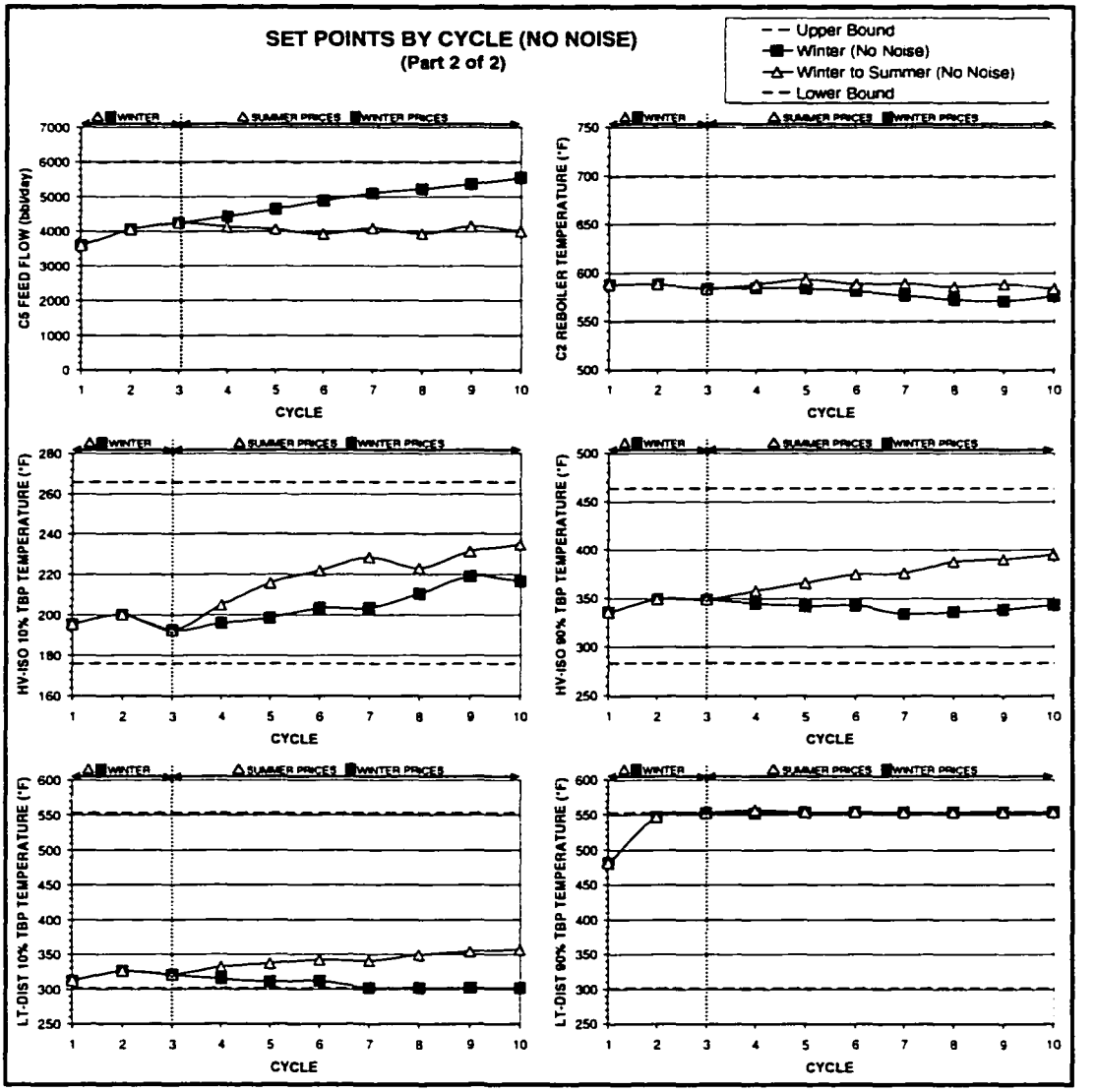


Figure 6.12: Graphs of the set point changes made during the winter case and winter to summer price change with no measurement noise runs using the ASPEN PLUS version 9.3 real time optimizer (2 of 2).

6.4 Process Disturbances

To test the effect of process disturbances on the performance of the real time optimizer, the example of a restriction on the cooling capacity of the condensers on columns C1 and C2 was chosen. This was implemented by adjusting the measured condenser temperatures returned from the "plant" simulation. For the cases tested, the condenser temperatures were forced to be 100°F, approximately 10°F higher than normal. This test was used to simulate an increase in ambient temperature and humidity around the plant, thereby increasing the temperature of the cooling water used in the condensers on columns C1 and C2.

Graphs comparing the profit objective functions at each cycle for the base case and condenser temperature disturbance runs, with and without measurement noise, are presented in Figure 6.13. Both runs, with and without measurement noise, were run for as many cycles as possible. For the the condenser disturbance run with measurement noise, the SQP optimization algorithm failed due to violated inequality constraints on the fifth cycle. The constraints violated were the upper limits on the C1 reflux flow and the Light Distillate stream 90% volume distilled true boiling point temperature. The error given by the optimization algorithm was "line search failed on final iteration: solution may not be optimal".

Similarly, the run with no measurement noise failed on the sixth cycle of the real time optimizer due to violated inequality constraints and a convergence failure within tear stream block on the stripper column. The constraints violated in this case were the upper limits on the Light Iso stream Reid vapour pressure and the C_4-Distillate stream C_5 mole fraction. The tear stream convergence failure appeared to be caused by problems with the convergence of the PETROFRAC flowsheet block, C2C4C5, during some of the tear stream iterations. This prevented the convergence of the tear stream block within the required 30 tear stream iterations. The error given by the optimization algorithm was "no feasible point found to quadratic subproblem".

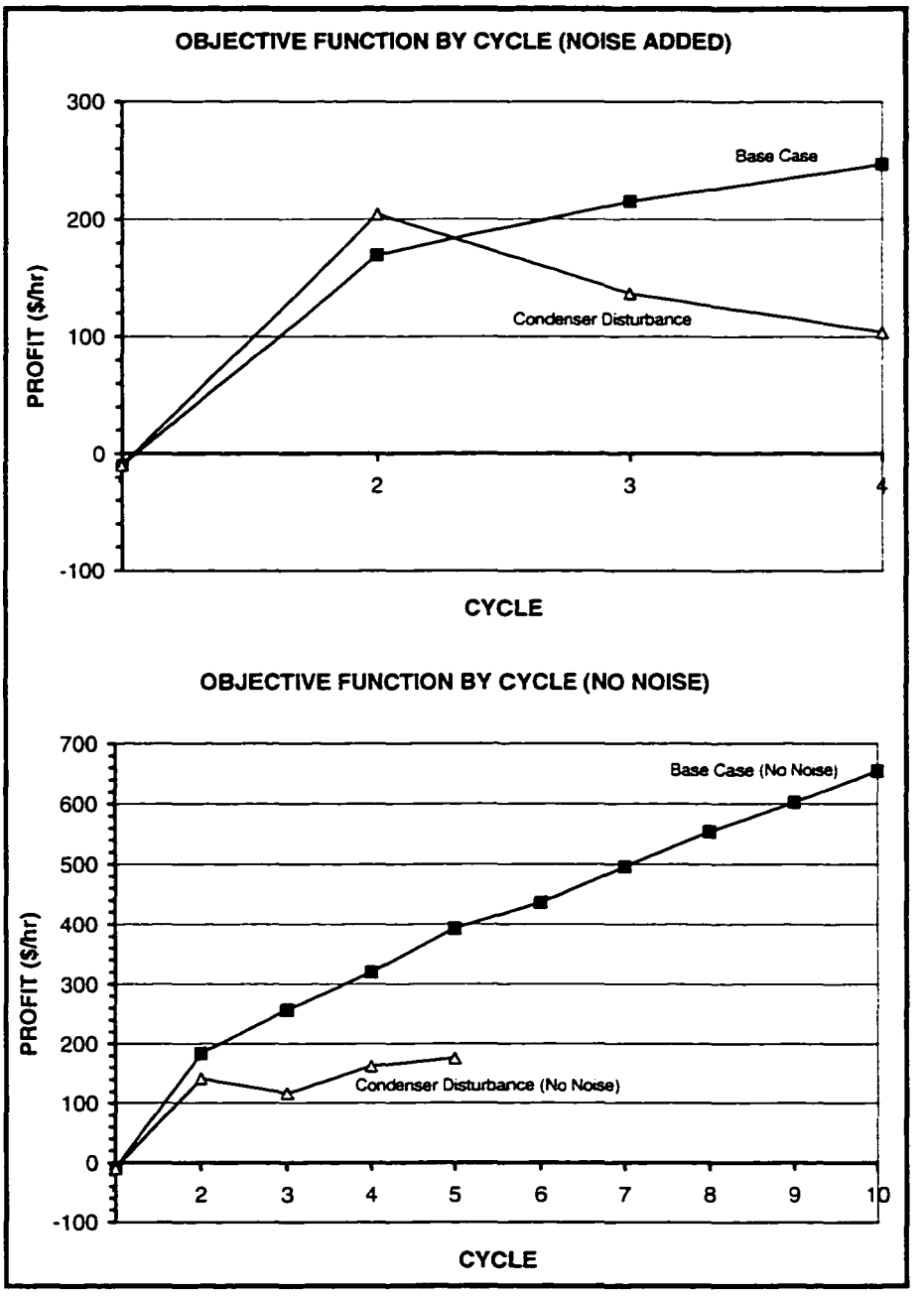


Figure 6.13: Graphs comparing profits (economic objective function) for base case and condenser disturbance runs with and without noise using the ASPEN PLUS real time optimizer.

The active inequality constraints during the condenser disturbance run with measurement noise were then same as for the base case run (see Section 6.2). For the condenser disturbance run without measurement noise, the active inequality constraints for cycles 2, 3 and 5 were the upper limits on the Light Iso stream Reid vapour pressure, the C_4-Distillate stream C_5 mole fraction and the Light Distillate stream 90% volume distilled true boiling point temperature. At the end of cycle 4, both the Reid Vapour pressure and C_5 mole fraction constraints were active. Figures 6.14 and 6.15 compare the changes in set points made at the end of each cycle for the base case and condenser disturbance runs without measurement noise added to the process data. For more information on the performance of real time optimizer on the two condenser disturbance runs, see Tables A.12 and A.13 in Appendix A.

6.5 ASPEN PLUS Software Upgrade

Soon after the previous real time optimizer runs were completed, the ASPEN PLUS process simulation package was upgraded from version 9.2 to 9.3. To test the effect of this upgrade, the real time optimizer was run without any modification using the new version of ASPEN PLUS on the base case. It was found that the optimizer failed on the second iteration due to process flowsheet convergence failures in both column C1 (modelled using RADFRAC) and columns C2, C4 and C5 (modelled using PETROFRAC). From the ASPEN PLUS version 9.3 manual [15], it was found that modifications had been been made to the RADFRAC and PETROFRAC unit operation models, and the SQP optimization algorithm.

During testing of the real time optimization problem, it was found that RADFRAC was now allows the use of different initialization methods for the column. It was found that setting the option `INIT-OPTION=CRUDE` in the RADFRAC model prevented the convergence errors encountered previously. No other changes were made to the real time optimizer code.

The graph in Figure 6.16 compares the profits returned from the economic

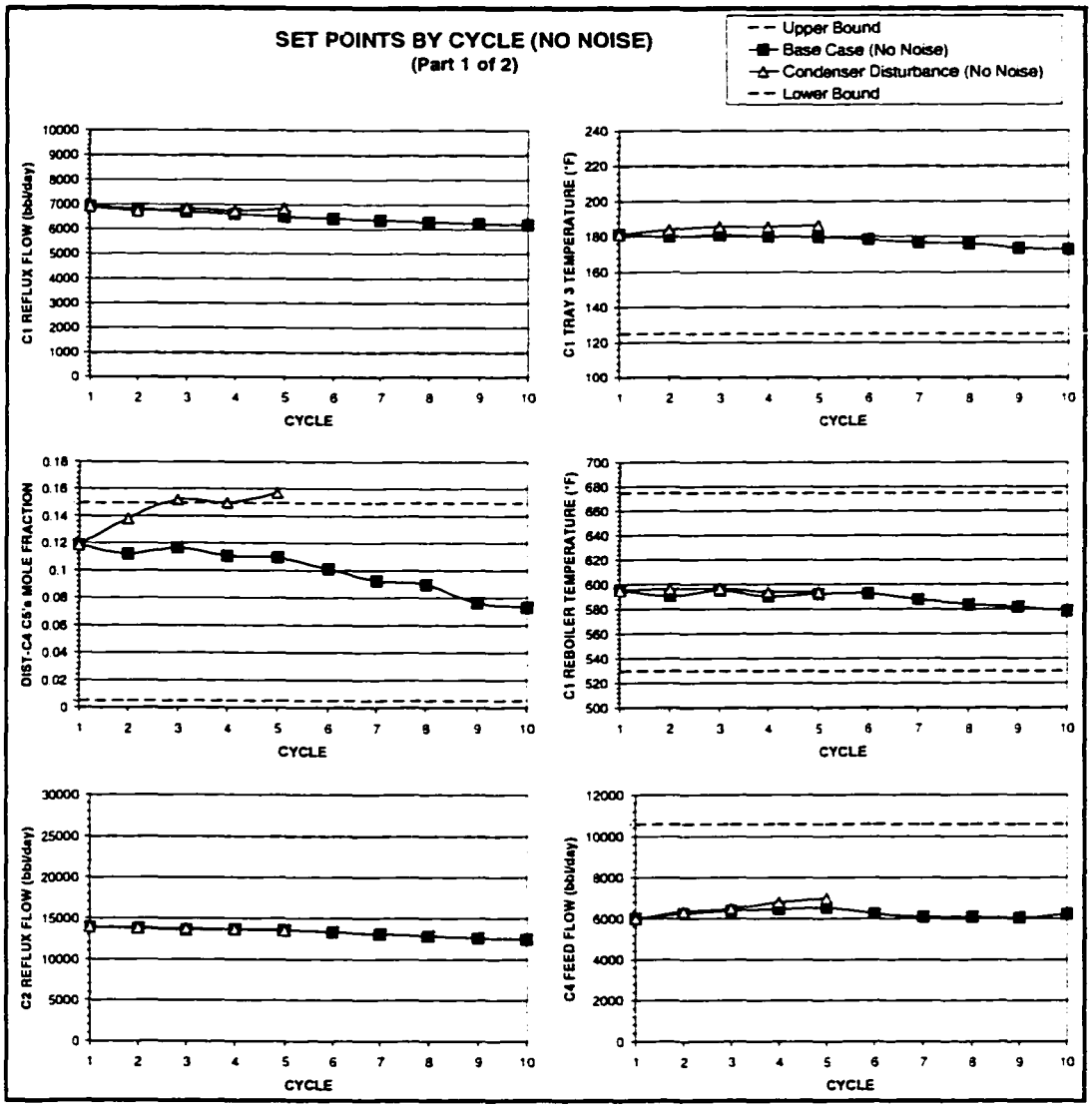


Figure 6.14: Graphs of the set point changes made during the base case and condenser disturbance with no measurement noise runs using the ASPEN PLUS real time optimizer (1 of 2).

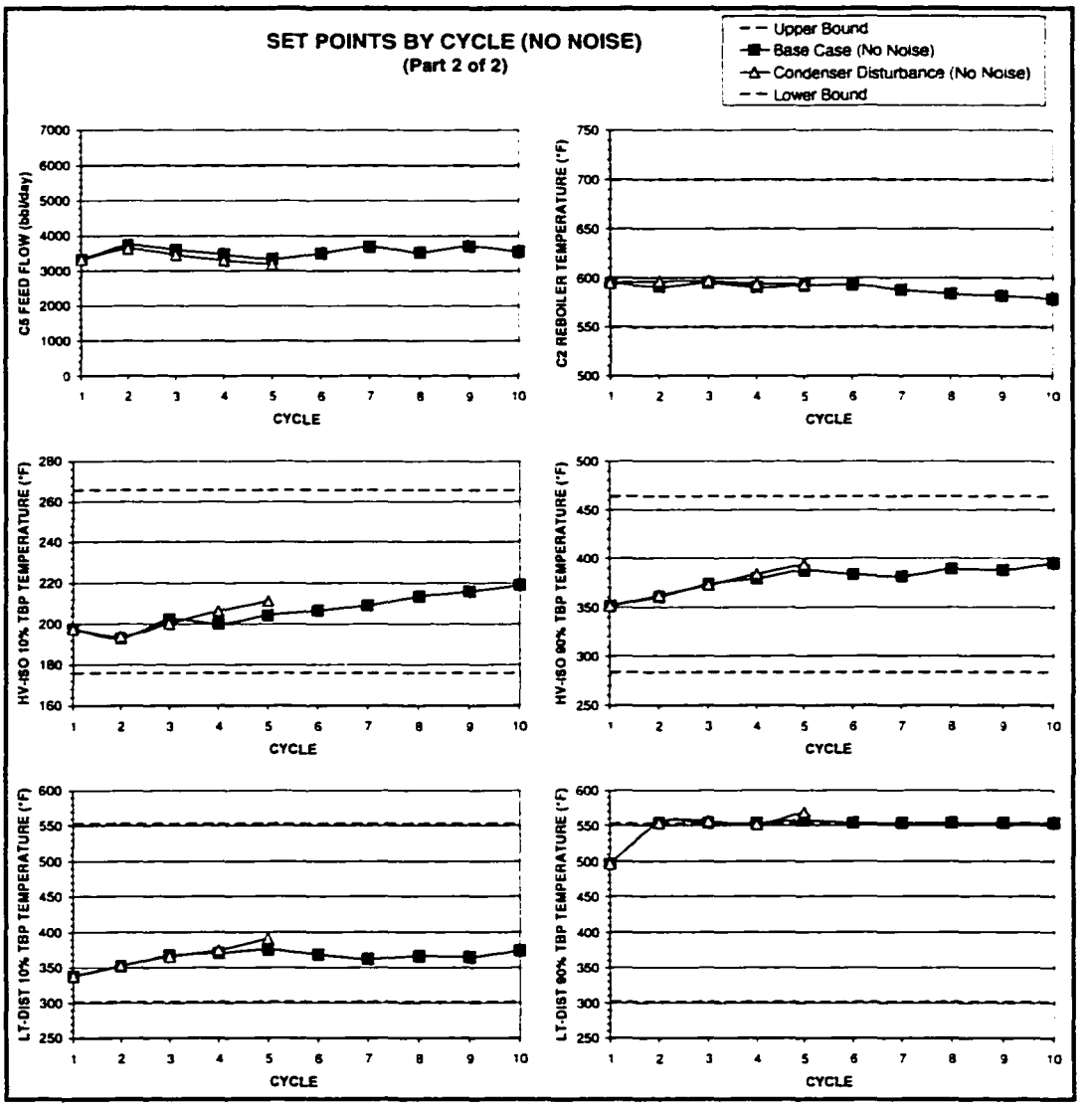


Figure 6.15: Graphs of the set point changes made during the base case and condenser disturbance with no measurement noise runs using the ASPEN PLUS real time optimizer (2 of 2).

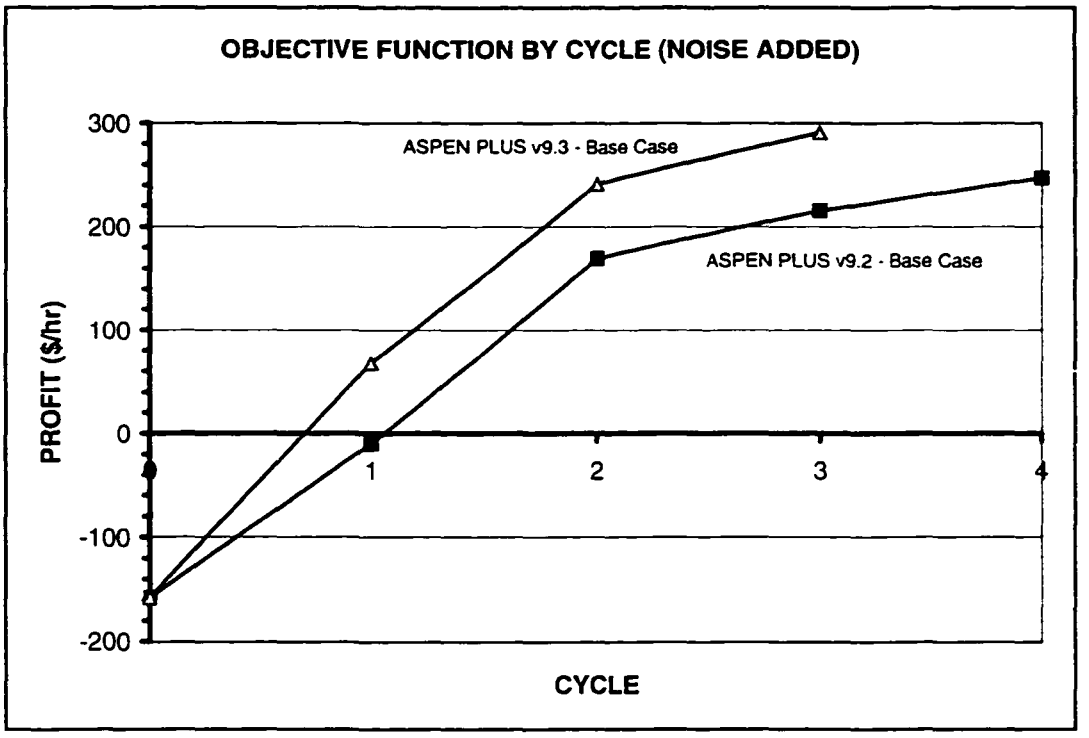


Figure 6.16: Graph comparing base case profits (economic objective function) with measurement noise for ASPEN PLUS versions 9.2 and 9.3

optimization for the two versions of ASPEN PLUS for the base case with measurement noise added to the plant data. The ASPEN PLUS version 9.3 real time optimization failed on the fourth cycle due to a violated inequality constraint which the SQP optimization algorithm could not satisfy. The violated constraint was the upper limit on the Light Distillate stream 90% volume distilled true boiling point temperature. The error given by the optimization algorithm was "line search failed on final iteration; solution may not be optimal". For more information on the performance of the ASPEN PLUS version 9.3 real time optimizer, see Table A.14 in Appendix A.

6.6 Discussion

6.6.1 Real Time Optimizer Linking

Formation of the complete real time optimizer was relatively simple for both the sequential modular and equation-oriented process simulators. The linking of the real time optimizer took 37.5 hours of development time for ASPEN PLUS and 46 hours for SPEEDUP. Including the times for developing the process model, data reconciliation and economic optimization, the total real time optimizer development times were 436.5 hours for ASPEN PLUS and 886 hours for SPEEDUP.

The ability to perform sequencing of the data reconciliation and economic optimization simulations in ASPEN PLUS greatly simplified the linking of the real time optimizer. This allowed the whole real time optimization to be performed with a single ASPEN PLUS input file using one process flowsheet with two optimization blocks sequenced to run consecutively. SPEEDUP does not have a sequencing feature, therefore the data reconciliation and economic optimization simulations had to be performed using separate input files, and were sequenced externally using the Unix C-Shell controlling the real time optimizer.

SPEEDUP's ability to save the present operating point of a simulation and to re-read this operating point back into the simulation at any time simplified the transfer of data between simulations. In ASPEN PLUS, the transfer of the simulations settings at the end point of the economic optimization to the data reconciliation for the beginning of the next cycle had to coded manually using a user defined FORTRAN block. This was time consuming and required debugging of the FORTRAN code.

6.6.2 Real Time Optimizer Comparison

Both the sequential modular and equation-oriented process simulator based real time optimizers were able to be compared on only two problems due

to difficulties tuning the objective function weights in SPEEDUP. The two problems tested were the base case, using the economic data given in Table 5.1, and the base case with no measurement noise added to the data produced from the "plant" simulation.

Figure 6.3 shows that both the SPEEDUP and ASPEN PLUS real time optimizers were reasonably close to the expected global optimum when they failed for the case where no random noise was added to the measured data (SPEEDUP finished \$53.30/hr and ASPEN PLUS \$97.14/hr lower than the expected global optimum). However in Figure 6.6, for the runs where random noise was added to the process measurements, both optimizers were unable to get close to the expected global optimum (SPEEDUP finished \$372.43/hr and ASPEN PLUS \$504.92/hr lower than the expected global optimum). The global optimum shown in Figures 6.3 and 6.6 cannot be guaranteed to be the true economic global optimum for this process due to the highly complex and nonlinear nature of the objective function, and also because of any process modelling errors that may exist.

The graphs in Figures 6.3 and 6.6 show that SPEEDUP was able to achieve significantly higher profits than ASPEN PLUS, even though less optimizer cycles were able to be completed. The most likely reasons for the more aggressive economic optimization performed by SPEEDUP is due in part to the more accurate derivative information used by the optimization algorithm from the process model, and the use of an infeasible path optimization algorithm. Equation-oriented process models are able to produce analytical derivative information, where possible, instead of the numerical derivative information calculated from a sequential modular process flowsheet. This not only reduces the computational time required for an optimization, it also provides more accurate information for the optimization algorithm to estimate step lengths and search directions. The infeasible path SRQP optimization algorithm does not require the process flowsheet to be converged at every iteration and permits violation of constraints and variable bounds on the way to the optimum. ASPEN PLUS's SQP optimization algorithm can

only violate the inequality and equality constraints specified in the input file constraint blocks at each iteration. This ability to violate constraints allows the optimizer the possibility of finding a better optimum by stepping outside constraints temporarily on the way to the optimum.

Also, SPEEDUP used significantly less computational time to complete each real time optimization cycle requiring an average of 232.14 CPU minutes for the base case and 93.70 CPU minutes for the base case with no measurement noise. ASPEN PLUS required an average of 479.11 CPU minutes for the base case and 116.10 CPU minutes for the base case with no measurement noise to complete a real time optimizer cycle. The main reason for this difference in computational times is the iterative method used by the sequential modular simulator to solve tear stream blocks. See Section 4.4.1 for a more thorough discussion of how tear stream convergence affects the computational time taken by ASPEN PLUS.

As already discussed in Sections 4.4.2 and 5.4.2, many problems were encountered tuning the SPEEDUP real time optimizer by adjusting weighting factors in the objective function to promote convergence of the SRQP optimization algorithm. Unfortunately, the SRQP algorithm does not allow the user to modify any parameters that affect the algorithm's criteria for "slow convergence", the most common error message returned when the optimization fails to converge. Another factor that may contribute to the convergence difficulties in SPEEDUP optimization problems is poor flowsheet and manipulated variable scaling with the process models used. At this time, automatic flowsheet scaling using user defined scaling parameters is only available for steady state and dynamic simulations. Flowsheet scaling for optimization problems will be added in a future version of SPEEDUP according to the program documentation [11, 12]. Flowsheet scaling was observed to have significant benefits in promoting flowsheet convergence during the steady state development of the process model (see Section 3.5.3).

Another problem encountered with the equation-oriented simulation based real time optimizer was the failure of the economic optimization due to prop-

erty errors during the third cycle of the base case run. This error in an undetermined flash calculation occurred consistently on the second iteration of the economic optimization on the third cycle of the real time optimizer. This error did not prevent the simulation from continuing, but did cause the constraint violations to be approximately 3 orders of magnitude larger than normal during this optimization. This resulted in the SRQP optimization algorithm failing due to "inconsistent constraints" on every attempt to solve this problem. Some method of tracking which flash and/or model equations caused this property error is required to allow the user a chance to place bounds on the offending variable to restrict its movement within reasonable limits.

Several problems were also encountered with the sequential modular simulation based real time optimizer. The most common problem was the inability of the SQP optimization algorithm to satisfy violated constraints after converging the economic objective function. This usually occurred when two or more constraints became active during the economic optimization. The two inequality constraints which commonly caused the failure of the ASPEN PLUS real time optimizer by being violated were the upper limit on the Light Distillate stream 90% volume distilled true boiling point temperature, and the upper limit on the C_4-Distillate stream C_5 mole fraction. By looking at the optimization history, it appears that once a constraint is violated, the SQP optimization algorithm makes very little effort to satisfy the constraint until the end of the optimization. At this stage, the constraint has usually been violated to such an extent, that the optimizer is unable to satisfy the constraint on subsequent iterations. It is likely that this problem is caused by the lack of accurate derivative information available to the SQP optimization algorithm from the sequential modular process flowsheet.

Occasionally, failures in the tear stream convergence were noticed during flowsheet passes of the ASPEN PLUS process flowsheet. These failures generally did not cause any problems other than adding extra computation time to the optimization convergence, and possibly returning inaccurate numerical

derivative information to the SQP optimization algorithm. Flowsheet convergence was only found to be a problem in one run of the ASPEN PLUS real time optimizer. It occurred on the sixth cycle of the condenser disturbance with no measurement noise run (see Section 6.4). However, since this process flowsheet contains relatively few recycle loops, convergence of the flowsheet recycle loops is a concern when using sequential modular process simulators in real time optimization on more complicated industrial processes.

From the graphs in Figures 6.4 and 6.5 comparing the set point changes made by both real time optimizers on the base case without measurement noise, it can be seen that the SPEEDUP tended to make fairly large changes to the composition related set points. In particular, the largest changes occurred on the C_4-Distillate stream C_5 mole fraction and the Heavy Iso stream 10% volume distilled true boiling point temperature. Decreasing the C_4-Distillate stream C_5 mole fraction as low as possible increases the profit by forcing these C_5 components into the higher priced Light Iso product stream, provided the constraints on the bubble point in the condenser on column C2 and the Light Iso stream Reid vapour pressure can be maintained. ASPEN PLUS is slower to accomplish this reduction in C_5 components in the C_4-Distillate product stream. In turn, the Heavy Iso product stream 10% volume distilled true boiling point temperature is increased to provide some heavier components in the Light Iso stream to reduce the Reid vapour pressure. Reducing the Reid vapour pressure increases the Light Iso stream price because of the quality specification associated with the Light Iso Reid vapour pressure as given in Equation (5.3). It is likely that the SPEEDUP real time optimizer made these changes in set points more quickly than ASPEN PLUS because of SPEEDUP's infeasible path optimization method, and also due to the more accurate derivative information provided by the process flowsheet in an equation-oriented simulation. Both real time optimizers quickly increased the Light Distillate stream 90% volume distilled true boiling point temperature the upper limit, to allow as many heavy components as possible into the Light Distillate product stream rather than to the splitter bottoms

stream for further cracking. It is likely that this constraint will always be active during this economic optimization for this reason.

Figures 6.4 and 6.5 also show that at the end of the first cycle, there were already significant differences in the C_4-Distillate stream C_5 mole fraction between the two real time optimizers. Some of this difference can be attributed to the more aggressive optimization performed by SPEEDUP's SRQP optimization algorithm. However, this large difference may have also been caused in part by differences in the results from the data reconciliation of the measured data performed by both of the real time optimizers. Slight differences between the process models used in the two process simulation packages may also contribute to this difference.

6.6.3 Economic Model Changes

The first economic runs performed tested the effect of changes in the Heavy Iso and Light Distillate product stream prices. These runs were chosen to test the performance of the ASPEN PLUS real time optimizer under different economic conditions where the expected performance of the optimizer was known from Shell Canada's experience. In particular, the base case run where the price of the Heavy Iso stream was \$193.89/m^3 and the Light Distillate stream was \$180.52/m^3, and the run where the price of the Heavy Iso stream was \$173.89/m^3 and the Light Distillate stream was \$200.52/m^3, correspond to typical summer and winter economic operating conditions respectively (however, the summer or base case has the same Light Iso Reid vapour pressure upper limit of 14 psi as used under normal winter operation). The graphs of the objective function profit at the end of each real time optimizer cycle given in Figure 6.7 show similar shaped curves for all of the runs with different stream prices.

Differences in the performance of the real time optimizer in the runs with different Heavy Iso and Light Distillate stream prices can be seen on the graphs of the set point changes made in Figures 6.8 and 6.9. Not surprisingly, the runs where the Heavy Iso and Light Distillate stream prices were both

at their highest values or at their lowest values produced very similar set point changes as the difference between the two stream prices was the same for both runs. The largest difference in set point changes made were for the base case (summer) run and the winter run. For the base case run, the emphasis for the economic optimization is to produce as much Heavy Iso product as possible, because this stream has the highest associated price. To achieve this, the optimization forces the set points on the Heavy Iso stream 90% volume distilled and the Light Distillate stream 10% volume distilled true boiling point temperatures as high as possible to encourage the production of as much Heavy Iso as possible. Also the flow to the column C5 sidestripper, where the Light Distillate product draw is taken, is reduced as much as possible within the process constraints.

Similarly for the winter run, the emphasis for the economic optimizer is to produce as much Light Distillate product as possible. To achieve this, the optimization forces the set points on the Heavy Iso stream 90% volume distilled and the Light Distillate stream 10% volume distilled true boiling point temperatures as low as possible to encourage the production of as much Light Distillate as possible. These results are similar to those observed with Shell Canada's real time optimizer.

The results of the sensitivity study on the economic objective function performed by Fenton [39] in Table 6.1 seem to show some inconsistent data. The optimizations performed with the modified economic data were all initialized with the results of the optimized base case run. However, it is likely that the optimized base case was not solved to the global optimum profit, and that the optimizer on this run finished at a local optimum of the profit objective function. This is especially evident from the high and low fuel oil costs where the optimizer was able to increase the starting profit by 31% and 47% respectively. This seems to represent a large change in optima for just a $\pm 5\%$ change in the reboiler furnace heating oil cost. Unfortunately, there is no way to guarantee that an optimization has reached the global optimum of a given objective function, therefore is is always possible that the profit

returned from the real time optimization is actually a local optimum.

This economic sensitivity study does, however, show how continuous optimization of the process can help increase the profit made by the process. In every run tested in this study, the economic optimization was able to find a better operating point. Therefore, even though the optimum found cannot be guaranteed to be the global optimum for the process, it can be seen that continuously optimizing the plant does produce higher profits than not optimizing after changes in economic parameters.

The effect of changing the economics from typical winter to typical summer operating conditions part way through the optimizer run can clearly be seen in Figure 6.10. For these runs, the Light Iso Reid vapour pressure upper limit was also changed for summer to 10.5 psi from the limit of 14 psi used during winter operation. Changing the economics at the beginning of cycle 3 for the run with measurement noise and cycle 4 for the run with no measurement noise caused the profits to remain almost constant for three cycles as the real time optimizer changed optimization strategies. The optimizer was then able to make gains in profit for the run with no measurement noise at almost the same rate as the winter case. This was an expected result. as the optimizer had been trying to maximize the production of Heavy Iso up to the change in economic conditions, and had to switch to maximizing production of the Light Distillate product stream. This can be seen in the graphs comparing the set point changes made at the end of each cycle for the runs with no measurement noise added, given in Figures 6.11 and 6.12. For the winter case, the real time optimizer lowered the Heavy Iso product stream 90% volume distilled and the Light Distillate stream 10% volume distilled true boiling point temperatures as much as possible on each cycle, to allow as many components as possible to be included in the Light Distillate product. For the summer cycles of the winter to summer run, the real time optimizer swapped to a strategy of increasing the Heavy Iso product stream 90% volume distilled and the Light Distillate stream 10% volume distilled true boiling point temperature to maximize the amount of Heavy Iso being

produced for gasoline products.

The change in the Light Iso Reid vapour pressure upper constraint from 14 psi to 10.5 psi had an obvious effect on the operation of the condenser on the C1 stabilizer column. From Figure 6.11, it can be seen that the amount of C_5 components retained in the C_4-Distillate stream increased noticeably. This is to reduce the number of lighter components in the Light Iso product, thereby lowering the Reid vapour pressure of this stream to comply with the lowered Reid vapour pressure constraint.

6.6.4 Process Disturbances

The disturbance created by the increased condenser temperatures in both columns C1 and C2 reduced the profit attained by the real time optimizer significantly, as can be seen in Figure 6.13. Both of the condenser disturbance runs with and without measurement noise produced similar results, with profit objective function unable to be pushed higher than approximately $180/hour. For the first two cycles of both condenser disturbance runs, the profits obtained were almost identical to the base case runs at which point the optimizers were unable to make any further significant gains in profit.

Comparing the set point changes made during the base case and condenser disturbance runs with no measurement noise in Figures 6.14 and 6.15, it can be seen that the major difference was in the handling of the C_4-Distillate C_5 mole fraction set point. This set point was forced to its upper bound within 2 cycles of the real time optimizer to reduce the amount of light components reaching the splitter column This was the expected reaction to a loss in cooling in the condensers, therefore causing a reduction in the ability to condense lighter components at the top of both columns. This resulted in increased Vapour product stream flow and a corresponding decrease in C_4-Distillate product stream flow compared with the base case run.

Originally this run was planned have the process disturbance added at the ASPEN PLUS base case optimum solution. However, due to the problems the

SQP optimization algorithm had in satisfying active inequality constraints once the process disturbance was added, this was not possible. Therefore, it was decided to have the process disturbance added right from the start of the run. Unfortunately this meant that no runs were possible to test the performance of the real time optimizers once it had attained a near optimal operation of the plant.

6.6.5 ASPEN PLUS Software Upgrade

From the results shown in Figure 6.16, it can be seen that even small changes to process models and optimization algorithms can have a significant effect on the performance of a real time optimizer. The optimization algorithm in ASPEN PLUS version 9.3 was able to obtain a slightly higher profit (approximately $40/hr more) in fewer real time optimizer cycles. This profit was still approximately $100/hr less than the profit achieved by SPEEDUP on the base case, and used more real time optimizer cycles and computation time. The ASPEN PLUS version 9.3 real time optimizer still failed due to violated inequality constraints in a similar manner to the version 9.2 real time optimizer.

The problems encountered in trying to achieve convergence of the real time optimizer with the new version of ASPEN PLUS show how care must be taken when upgrading process simulation packages. Testing, debugging and retuning of the real time optimizer needs to be performed to verify that the real time optimization application is still producing valid results.

NOTE TO USERS

Page(s) missing in number only; text follows. Microfilmed as received.

162

UMI

Chapter 7

Conclusions

The conclusions from this research are covered in the next three sections. Conclusions from the development of the real time optimizers using both sequential modular and equation-oriented process simulation packages are covered in Section 7.1 below. Comparison of the two types of process simulation packages used with respect to their performance in real time optimization applications are covered in Section 7.2. In Section 7.3, conclusions specific to the stabilizer-splitter process studied are made. Finally, in Section 7.4, some suggestions for future work following on from this research are made.

7.1 Real Time Optimizer Development

Overall, it can be concluded that sequential modular process simulation packages are easier to use when developing real time optimization applications, however equation-oriented simulators are more flexible for manipulated variable choice and recycle loop convergence. Development of the same real time optimizer using SPEEDUP, an equation-oriented process simulator, took approximately twice as long as the development using ASPEN PLUS, a sequential modular process simulator.

The main reason for this development time difference was due to the lack of useful, or meaningful, error messages produced by an equation-oriented

simulator when either the process flowsheet or optimization algorithm fails to converge. Debugging convergence problems is a slow process of eliminating all possible sources of error over many runs of the simulation, since different errors often result in the same error message from the convergence algorithm. This lack of useful information is caused in part by the simultaneous solution of equation-oriented process flowsheet equations and also the infeasible path convergence algorithms used. Since an equation-oriented simulator solves the process flowsheet as a large set of simultaneous equations, it is difficult to track errors to a particular equation or variable.

A few mathematical tools are available in SPEEDUP to help determine the cause of a convergence failure. One method of tracking these errors is to monitor the largest equation residuals during the solution of the flowsheet to track down problem equations and/or variables. Another method is to track the variables which changed by the largest amount between iterations of the flowsheet convergence algorithm. These equations and/or variables can then be checked by hand to see if the problem is caused by incorrect modelling equations, poor starting point values, incorrect variable bounds, or is caused by errors in other variables or equations. However, the large number of equations and variables involved, 6066 equations and unknown variables for the stabilizer-splitter process flowsheet, make this a time consuming task. The infeasible path algorithms used to solve the process flowsheet and optimization problems also make it difficult to detect errors because intermediate variable values do not necessarily satisfy constraints and bounds. This can result in some strange intermediate variable values which have no physical meaning, for example, negative component compositions. Therefore looking at the actual intermediate value of variables provides very little information with respect to convergence failures.

Sequential modular process flowsheet errors can always be tracked to a particular unit operation, and often a reasonably accurate error message is produced. This makes debugging of flowsheet convergence problems relatively simple. Optimization algorithm convergence failures are somewhat

more difficult to debug, however more useful information is available for finding the cause of the error than in an equation-oriented simulator. Since the flowsheet solution is always feasible at the end of an iteration, variable values can be used to help find the cause of the convergence failure.

Sequential modular process simulators usually also have the advantage of a graphical user interface, however, graphical user interfaces are presently being developed for several equation-oriented packages including SPEEDUP and MASSBAL. A graphical user interface is useful for preventing syntax errors in the input file, quickly building process flowsheets and ensuring that the correct number of variables have been specified before running a simulation. However most of the time in developing a real time optimizer is spent fixing convergence errors and ensuring that the process flowsheet is adequately modelling the plant. For these tasks, a graphical user interface does not offer many advantages over direct editing of the input code.

An equation-oriented process simulator has two main advantages over a sequential modular process simulator when developing a real time optimizer. The first advantage is how equation-oriented process simulators handle the convergence of recycle loops and tear streams. A recycle loop in an equation-oriented simulator is handled by inserting a tear block which equates the inlet and outlet streams plus a slack variable which is set to zero. The tear stream is then converged at the same time as all the other flowsheet equations. In a sequential modular simulator, a recycle loop results in the creation of a tear stream convergence loop which iterates through all the blocks in the recycle loop until convergence of the torn stream is obtained. This tear stream convergence method is computationally time consuming and also less robust, particularly for large recycle loops. For complex process flowsheets with more recycle than the stabilizer-splitter flowsheet studied, convergence of tear streams in a sequential modular simulator would be extremely difficult to guarantee over the range that the real time optimizer is required to operate.

The second main advantage of using equation-oriented process simulators to develop a real time optimizer is their flexibility in choosing specified or

manipulated variables. A sequential modular simulator restricts the user to specifying the inputs available for that process model. If any other variables need to specified, a design specification is required which is solved using an iterative procedure to converge the specification. In an equation-oriented simulator, any combination of variables (input or output variables) can be chosen directly as the specified or manipulated variables, provided they fulfil the degrees of freedom required, and do not over or under specify an equation block after decomposition has been performed. This does not mean that any combination of variables will necessarily converge the flowsheet, but it does offer more flexibility than a sequential modular flowsheet.

This flexibility in the choice of manipulated variables does cause some other problems. During the development of the SPEEDUP real time optimizer, problems were encountered when the manipulated variables chosen fully specified the mass balance around column C1 and the energy balance around column C2. Since this problem was not detected during analysis of the flowsheet by SPEEDUP, solution of the flowsheet was performed which resulted in many different error messages such as "inconsistent constraints", "slow convergence" and various property errors. This problem was only detected after extensive analysis of the process flowsheet by hand. For larger flowsheets, this problem would be difficult and time consuming to find and correct.

Several process simulator specific problems were found with both ASPEN PLUS and SPEEDUP. Errors in the transfer of data between blocks in ASPEN PLUS using FORTRAN common blocks were difficult to debug since there is no automatic method for checking the variable values being transferred. Also ASPEN PLUS has no simple mechanism for saving the results of a simulation, and then using these results to initialize another simulation as with SPEEDUP. Several useful features not available in SPEEDUP at present are the addition of variable scaling for optimization simulations, sequencing of global optimization blocks similar to ASPEN PLUS's convergence block sequencing, and improved error tracking for property errors indicating in

which unit the error occurred.

When developing the real time optimizers using both types of process simulator, it was found that the properties used had a large effect on the accuracy of the process models. In particular, it was found with the stabilizer-splitter process that the cross over from conventional components to pseudocomponents was critical when modelling the operation of the condensers in both columns C1 and C2. When analysing the assay data provided, if was found that PROPERTIES PLUS assumed the stream was a typical refinery stream, and therefore used typical specific gravity, boiling point and molecular weight curves when estimating the pseudocomponent properties. However, hydrocracked refinery streams exhibit different typical stream properties, particularly for the light and heavy pseudocomponents, which had to be corrected. This problem highlights how errors can be caused by not checking the underlying assumptions used in the property calculation routines.

7.2 Real Time Optimizer Comparison

From testing of both the equation-oriented and sequential modular process simulator based real time optimizers, it can be concluded that the equation oriented real time optimizer was a far more powerful optimizer. This was due to the benefits of open form equation solution methods allowing for very robust and quick tear stream convergence, infeasible path optimization methods not requiring the solution of the flowsheet on each iteration and the availability of more accurate analytical derivative information from the process model equations. The closed form equation based models, with only one recycle loop in the flowsheet, took approximately between 2 and 4 times longer to complete a cycle of the same optimization and data reconciliation problems as the open form equation based models. A more complicated flowsheet with more recycle would add even more to the convergence time for a sequential modular simulator, in addition to the problems with convergence robustness for closed form tear stream solution.

A process simulator specific conclusion that should be noted is that neither the ASPEN PLUS nor the SPEEDUP process simulator based real time optimizers were robust enough for an industrial application in their present form. SPEEDUP, however, was able to find better profits with less optimizer cycles and computational time than ASPEN PLUS on all runs tested. The following conclusions are based on the results from the testing of the two real time optimizers. Where appropriate, more general conclusions will also be pointed out.

Two problems were encountered when testing the ASPEN PLUS real time optimizer. The first was the poor handling of violated inequality constraints by the SQP optimization algorithm used in ASPEN PLUS. This was the most common reason for failure of the real time optimizer and almost always occurred when two inequality constraints had been active on the previous cycle of the optimizer. The economic optimizer would then violate one or both of these constraints searching for an improved profit objective function, however it was unable to satisfy these constraints again once a new optimum had been found. The most commonly violated constraints were the upper limits on the column C1 reflux flow rate and the Light Distillate stream 90% volume distilled true boiling point temperature. The most likely cause for this problem is the SQP optimization algorithm implemented in ASPEN PLUS. With more refinement of the algorithm, it is possible this problem may be alleviated. Another contributing cause for this is the less accurate derivative information available for the optimization algorithm from the numerical derivatives calculated using the sequential modular process flowsheet models. Adding to some of the poor derivative information, some flowsheet passes used to determine derivative values failed to converge due to tear stream or process model convergence failures. The less accurate derivative information is typical of all sequential modular flowsheet simulators due to the closed form of the modelling equations.

The second problem encountered using ASPEN PLUS was the occasional failure to converge the tear stream block. This failure usually did not di-

rectly cause failure of the real time optimizer, but did add significantly the computational time taken during optimization. It also possibly contributed to inaccurate information being passed to the SQP optimization algorithm. For complex flowsheets with more recycle loops than the stabilizer splitter process, tear stream convergence failures would make optimization of the process very difficult to achieve over all process operating conditions. Problems converging tear streams quickly and robustly is typical of all closed form equation solution methods because of the iterative nature of their solution.

The main problem with the SPEEDUP real time optimizer was the lack of tuning parameters available for the SRQP optimization algorithm. Almost all convergence failures encountered when running the SPEEDUP real time optimizer produced the error message "slow convergence", however there was no method for changing the criteria used to determine slow convergence. On many occasions, it appeared that the optimizer was very close to the optimum when the "slow convergence" error occurred. For this reason, weighting factors had to be used in the objective functions for the data reconciliation and economic optimization to promote convergence of the optimization algorithm. These weighting factors could not be automatically determined during the real time optimization and required the user to find their value which allowed convergence. This is not practical for implementation in a real time optimizer. Another factor which may have caused these convergence problems was scaling of the process flowsheet and manipulated variables. This result is a process simulator specific problem, as many other optimization algorithms are available which may be more suitable for solving this problem. Unfortunately, no other optimization algorithms were available for testing with SPEEDUP for this study.

Another problem encountered with SPEEDUP was the occurrence of property errors, particularly during flash calculations. SPEEDUP does not provide any method of tracking which process model flash procedure was being calculated at the time. This would allow bounds restricting movement of some variables to be set, preventing the optimizer from moving into regions

where property calculations may fail. Again, this is likely to be a process simulator specific result due to problems either in the properties or convergence algorithms used.

SPEEDUP consistently found better profit objective function values, usually by $200/hr or more, at each cycle of the real time optimizer than ASPEN PLUS. For the base case runs with no measurement noise (see Figure 6.3), both ASPEN PLUS and SPEEDUP came very close to achieving the expected global optimum before the real time optimizers failed. However, SPEEDUP used half as many cycles to achieve this increase in profit. The reason for this could be seen from SPEEDUP's more aggressive set point changes made at the end of each real time optimizer cycle. The most likely reasons for the better performance of the SPEEDUP real time optimizer are more accurate analytical derivative information available from the equation-oriented process flowsheet and the infeasible path optimization algorithm used.

The SPEEDUP real time optimizer was also computationally more efficient than ASPEN PLUS, taking approximately half as long to complete each cycle of the real time optimizer. The difference in computation times was mostly due to the extra time taken by ASPEN PLUS to converge the tear stream on each flowsheet pass, and the time taken to calculate the numerical derivatives for the optimization algorithm. It is likely that for complex flowsheets with more recycle loops, that a sequential modular process simulator would become even slower relative to an equation-oriented simulator. This was an expected result for an open equation model based optimization.

It is likely that if the above problems with the SPEEDUP equation-oriented process simulation package can be addressed, then SPEEDUP could become a very useful tool for developing and performing real time optimization. Despite the added difficulty in developing and converging an equation-oriented process flowsheet model, the added flexibility in terms of variable specification, robust and rapid tear stream convergence, and more powerful optimization algorithms make the equation-oriented process simulator a more

attractive tool for real time optimization. It is unlikely that the problems with the ASPEN PLUS process simulator's ability to converge flowsheet recycle loops quickly and robustly can solved easily using the sequential modular architecture. Some simultaneous-modular methods have been incorporated into ASPEN PLUS to speed convergence of tear streams, however it is unlikely that these will achieve the level of performance required for complex flowsheets used in real time optimization. For less complex flowsheets with little or no recycle, a sequential modular simulator provides a very simple method to develop a real time optimization application, provided the SQP optimization algorithm's convergence with respect to violated constraints can be improved.

A final conclusion regarding upgrading versions of the process simulation software should be noted. When ASPEN PLUS was upgraded from version 9.2 to 9.3, the real time optimizer failed to converge without adjustments to the process simulation code. Considering that version 9.3 was only a minor software upgrade, this shows that care must be taken when upgrading to ensure that the real time optimizer is still producing valid results. Even small changes to the process models and the optimization algorithm can have an impact on the performance of the optimization. This conclusion should apply equally to both sequential modular and equation-oriented simulator based real time optimizers. Upgrading to version 9.3 of ASPEN PLUS did cause the real time optimizer to be slightly more robust, however constraint handling and tear stream convergence problems were still observed similar to those seen with version 9.2.

7.3 Process Specific Conclusions

The development of real time optimizers with both ASPEN PLUS and SPEEDUP fulfilled the objective of developing a "test bed" for real time optimization to allow further research to be conducted. It should be noted that the process studied in this thesis is somewhat less complex than those typically used

with industrial real time optimizers. The process chosen was smaller than normal to allow the solution of a real time optimizer cycle to be completed within a reasonable period of time with the computing resources available. Therefore it could be argued that the objective of real time optimization on this process as implemented could also be achieved using a combination of advanced control and process insight. This is because only two major operating strategies were observed in the testing of this process. that is. a summer and a winter operating strategy based on the economic conditions. To implement some form of model based local optimizer would be the correct decision in an industrial situation if this was the whole process to optimized. However. for the purposes of researching real time optimization this process produces valid results.

The stabilizer-splitter process at Shell Canada's Sarnia refinery is included in a real time optimizer which also optimizes the operation of the catalytic cracking units. This addition of several extra unit operations to the real time optimizer introduces many different operating strategies and trade-offs that can be made between process units. These trade-offs between the operation of several units and changes in operating conditions are where the payback from installing a real time optimization application can be made. Also. with more unit operations included in a real time optimizer. less conflict is possible between the operating strategies used in different parts of the plant.

All of the following conclusions were based on testing of the ASPEN PLUS real time optimizer under varying economic conditions and process disturbances.

Several real time optimizer runs were carried out to check the performance of the real time optimizer under typical summer and winter economic operating conditions. It was concluded that the real time optimizer performed as expected by trying to maximize production of the Heavy Iso gasoline product stream during typical summer economic conditions. and maximizing Light Distillate heating oil product stream under typical winter economics. Two

runs were also performed where the economic conditions were changed from winter to summer part way through the real time optimizer run. These runs produced similar set point changes and results after the change in economics as the previous real time optimizer run completely with typical winter economic conditions.

From the results of the winter only and the winter to summer runs presented in Section 6.3.3, the two operating strategies that could be implemented using advanced control for optimizing only the stabilizer-splitter process would be:

1. For summer economics and process constraints, the objective would be to increase the Heavy Iso and Light Distillate 90% volume distilled true boiling point temperatures as close to their constraints as possible while maintaining the Light Iso stream Reid vapour pressure at its upper limit.

2. For winter economics and process constraints, the operating objective is to increase the Light Distillate 90% volume distilled true boiling point temperature and decrease the Light Distillate 10% volume distilled true boiling point temperature as close to their constraints as possible. The C_4-Distillate stream C_5 mole fraction should also be reduced as much as possible while the Light Iso stream Reid vapour pressure at its upper limit.

A sensitivity study performed by individually varying the economic objective function prices by $\pm 5\%$ concluded that continually optimizing the plant does produce higher profits than not optimizing after economic changes. In every case tested, the real time optimizer was able to find a better operating point after a change in economic parameters. However, this study also showed that there is no way to guarantee that the optimization has found a global optimum profit objective function which made the results of this study difficult to analyse.

Finally, a process disturbance in the condenser cooling water temperature was simulated by increasing the condenser temperatures returned in the measured data to the real time optimizer. The real time optimizer performed as expected to the reduction in the available cooling in the condensers which significantly reduced the profit attained by the economic optimization.

7.4 Recommendations for Future Research

Researching real time optimization is difficult due to the large amount of time required to develop, debug and test the real time optimizer code. The development of an industrial real time optimization application from this research allows many areas of real time optimization to be studied in the future. Several possible areas of future research are outlined below.

A new version of SPEEDUP, version 6.0, is expected before the end of 1997 [16] which may address some of the shortcomings found when using SPEEDUP for real time optimization. With the addition of flowsheet scaling, SRQP optimization interface enhancements and optimization manipulated variable scaling, it should be possible to fix the "slow convergence" problems encountered with the SPEEDUP real time optimizer. Further testing of the equation-oriented real time optimizer will then be possible, thereby providing more comparison runs with the sequential modular simulator based real time optimizer.

Shell Canada has modified the stabilizer-splitter plant over the last year adding a draw from column C2 which is fed back to column C1, thereby creating a flowsheet recycle loop. By making this process modification to both the ASPEN PLUS and SPEEDUP based real time optimizers, this would show the effect of adding a recycle loop to the flowsheet on both sequential modular and equation-oriented solution methods. This study could also be used to show how easily the process model in the two types of process simulators can be modified and maintained.

The simulation model used as the "plant" in this study should be im-

proved to allow more effective analysis of the real time optimization applications. Modelling of the process control used in the stabilizer-splitter process using a dynamic simulation would allow several studies to be researched:

- the effect of real time optimization set point changes on the stability and performance of the plant's process control.
- the cost of disturbing the plant by making continuous process control set point changes compared to the profit gained by moving the plant's operating point.

NOTE TO USERS

Page(s) missing in number only; text follows. Microfilmed as received.

176

UMI

Bibliography

[1] American Petroleum Institute, Washington, D.C. *API Technical Data Book — Petroleum Refining*, fifth edition, 1992.

[2] Y. Arkun and G. Stephanopoulos. Optimizing control of industrial chemical processes: State of art review. In *Proceedings of the 1980 Joint Automatic Control Conference, San Francisco, California*, volume 1, pages WP5–A. American Automatic Control Council, 1980.

[3] Y. Arkun and G. Stephanopoulos. Studies in the synthesis of control structures for chemical processes. Part V: Design of steady-state optimizing control structures for integrated chemical plants. *AIChE J.*, 27:779–793, 1981.

[4] Aspen Technology Inc., Cambridge, Massachusetts. *ASPEN PLUS Getting Started Series — Modeling Petroleum Processes*, release 9.2 edition, November 1995.

[5] Aspen Technology Inc., Cambridge, Massachusetts. *ASPEN PLUS Reference Manual — Volume 1: Unit Operation Models*, release 9.2 edition, November 1995.

[6] Aspen Technology Inc., Cambridge, Massachusetts. *ASPEN PLUS Reference Manual — Volume 2: Physical Property Methods and Models*, release 9.2 edition, November 1995.

[7] Aspen Technology Inc., Cambridge, Massachusetts. *ASPEN PLUS Reference Manual — Volume 3: Physical Property Data*, release 9.2 edition, November 1995.

[8] Aspen Technology Inc., Cambridge, Massachusetts. *ASPEN PLUS User Guide Volume 1*, release 9.2 edition, November 1995.

[9] Aspen Technology Inc., Cambridge, Massachusetts. *ASPEN PLUS User Guide Volume 2*, release 9.2 edition, November 1995.

[10] Aspen Technology Inc., Cambridge, Massachusetts. *SPEEDUP Library Manual*, release 5.5d edition, March 1995.

[11] Aspen Technology Inc., Cambridge, Massachusetts. *SPEEDUP User Manual Volume I*, release 5.5d edition, March 1995.

[12] Aspen Technology Inc., Cambridge, Massachusetts. *SPEEDUP User Manual Volume II*, release 5.5d edition, March 1995.

[13] Aspen Technology Inc., Cambridge, Massachusetts. *Using SRQP with SPEEDUP*, release 5.5d edition, March 1995.

[14] Aspen Technology Inc. The Aspen Leaf. Cambridge, Massachusetts. April 1996.

[15] Aspen Technology Inc., Cambridge, Massachusetts. *What's New in ASPEN PLUS Release 9.3*, release 9.3 edition, August 1996.

[16] Aspen Technology Inc. Modeling Insights. Cambridge, Massachusetts. May 1997.

[17] J. K. Bailey, A. N. Hyrmak, S. S. Treiber, and R. B. Hawkins. Nonlinear optimization of a hydrocracker fractionation plant. *Comput. Chem. Eng.*, 17:123–138, 1993.

[18] L. T. Biegler. Chemical process simulation. *Chem. Eng. Prog.*, 88(10):50–61, 1989.

[19] L. T. Biegler, I. E. Grossman, and A. W. Westerburg. A note on approximation techniques used for process optimization. *Comput. Chem. Eng.*, 9:201–206, 1985.

[20] B. S. Bossen, L. J. Christiansen, J. E. Jarvan, and R. Gani. Simulation, optimization and data-reconciliation of industrial chemical processes. *Chem. Eng. Res. Des.*, 72(3):376–381, 1994.

[21] J. F. Boston and H. I. Britt. A radically different formulation and solution of the single stage flash problem. *Comput. Chem. Eng.*, 2:109, 1978.

[22] M. W. Brown, P. L. Lee, G. R. Sullivan, and W. Zhou. A constrained nonlinear multivariable control algorithm. *Chem. Eng. Res. Des.*, 68(5):464–476, 1990.

[23] G. Bussani, M. Chiari, M. G. Grottoli, S. Pierucci, T. Faravelli, G. Ricci, and G. Gioventu. Application of data-reconciliation and optimisation procedure to hydrogen plant. *Comput. Chem. Eng.*, 19 Suppl.:S299–S304, 1995.

[24] B. D. Campbell. Optimizing software systems can be created successfully. *Hydrocarbon Process.*, 63(11):109–112, 1984.

[25] C. L. Chen and S. Macchietto. *Successive Reduced Quadratic Programming — SRQP User's Guide.* Imperial College, London, release 1.0 edition, September 1989.

[26] C. M. Crowe. Reconciliation of process flow rates by matrix projection part ii: The nonlinear case. *AIChE J.*, 32:616–623, 1986.

[27] C. M. Crowe. Recursive identification of gross errors in linear data reconciliation. *AIChE J.*, 34:541–550, 1988.

[28] C. M. Crowe. Observability and redundancy of process data for steady state reconciliation. *Chem. Eng. Sci.*, 44:2909–2917, 1989.

[29] C. M. Crowe. Tests of maximum power for detection of gross errors in process constraints. *AIChE J.*, 35:869–872, 1989.

[30] C. M. Crowe. The maximum-power test for gross errors in the original constraints in data reconciliation. *Canadian J. Chem. Eng.*, 70:1030–1036, 1992.

[31] C. M. Crowe, Y. A. Garcia Campos, and A. Hyrmak. Reconciliation of process flow rates by matrix projection. *AIChE J.*, 29:881–888, 1983.

[32] C. R. Cutler and R. T. Perry. Real time optimization with multivariable control is required to maximize profits. *Comput. Chem. Eng.*, 7:663–667, 1983.

[33] C. R. Cutler and B. L. Ramaker. Dynamic matrix control — a computer control algorithm. In *Proceedings of the 1980 Joint Automatic Control Conference, San Francisco, California*, volume 1, pages WP5–B. American Automatic Control Council, 1980.

[34] M. L. Darby and D. C. White. On-line optimization of complex process units. *Chem. Eng. Prog.*, 67(10):51–59, 1988.

[35] M. Darouach and M. Zasadzinski. Data reconciliation in generalized linear dynamic systems. *AIChE J.*, 37:193–201, 1991.

[36] P. L. Douglas, P. S. Fountain, G. R. Sullivan, and W. Zhou. Model base control of a high purity distillation column. *Canadian J. Chem. Eng.*, 72:1055–1065, 1994.

[37] T. F. Edgar and D. M. Himmelblau. *Optimization of Chemical Processes*. McGraw-Hill, New York, 1988.

[38] F. C. Fatora and J. S. Ayala. Successful closed loop real-time optimization. *Hydrocarbon Process.*, 71(6):65–68, 1992.

[39] T. Fenton. Sensitivity analysis of a process optimisation. BASc Research Project, University of Waterloo, 1997.

[40] R. Fletcher. *Practical Methods of Optimization.* John Wiley and Sons, Chichester, second edition, 1987.

[41] American Society for Testing and Materials. ASTM D323 - 94: Standard test method for vapour pressure of petroleum products (Reid method). In *1995 Annual Book of ASTM Standards*, volume 05.01, pages 136–143, Philadelphia, 1995.

[42] J. F. Forbes and T. E. Marlin. Design cost: A systematic approach to technology selection for model-based real-time optimization systems. *Comput. Chem. Eng.*, 20:717–734, 1996.

[43] J. F. Forbes, T. E. Marlin, and J. F. MacGregor. Model accuracy requirements for economic optimizing model predictive controllers - the linear programming case. In *Proceedings of the 1992 American Control Conference, Chicago, Illinois*, volume 2, pages 1587–1593. American Automatic Control Council, 1992.

[44] J. F. Forbes, T. E. Marlin, and J. F. MacGregor. Model adequacy requirements for optimizing plant operations. *Comput. Chem. Eng.*, 18:497–510, 1994.

[45] P. W. Gallier and T. P. Kisala. Process optimization by simulation. *Chem. Eng. Prog.*, 83(8):60–66, 1987.

[46] C. E. Garcia and M. Morari. Internal model control. 1. a unifying review and some new results. *Ind. Eng. Chem. Process Des. Dev.*, 21(2):308–323, 1982.

[47] P. E. Gill, W. Murray, and M. Wright. *Practical Optimization.* Academic Press, New York, 1981.

[48] J. W. Hamer and C. B. Richenburg. On-line optimizing control of a packed-bed immobilised-cell reactor. *AIChE J.*, 34(4):626–632, 1988.

[49] M. Hanmandlu, P. Purkayastha, and J. K. Pal. On the use of nonlinear programming in real time control in process industries. In G. Di Pillo, editor, *Control Applications of Nonlinear Programming and Optimization — Proceedings of the Fifth IFAC Workshop, Capri, Italy.* IFAC, Pergamon Press, 1985.

[50] P. Harikumar and S. Narasimhan. A method to incorporate bounds in data reconciliation and gross error detection — ii. gross error detection strategies. *Comput. Chem. Eng.*, 17:1121–1128, 1993.

[51] Honeywell Hi-Spec Solutions, London, Ontario. *MASSBAL User's Guide Addendum*, July 1996.

[52] Honeywell Hi-Spec Solutions, London, Ontario. *OPTIMIZER Configuration Guide*, 1996.

[53] Honeywell Hi-Spec Solutions, London, Ontario. *OPTIMIZER for X Windows User Interface Guide*, 1996.

[54] A. Husain. *Chemical Process Simulation.* Wiley Eastern, New Delhi, 1986.

[55] Hyprotech Ltd. Inside Hyprotech. Calgary, Alberta, Spring 1997.

[56] S. Jang, B. Joseph, and H. Mikai. On-line optimization of constrained multivariable chemical processes. *AIChE J.*, 33:26–35, 1987.

[57] C. Kiparissides, G. Verros, and A. Pertsinidis. On-line optimization of a high-pressure low-density polyethylene tubular reactor. *Chem. Eng. Sci.*, 49:5011–5024, 1994.

[58] G. R. Kocis, G. R., and I. E. Grossman. Computational experience with DICOPT solving MINLP problems in process systems engineering. *Comput. Chem. Eng.*, 13:307–315, 1989.

[59] S. Krishnan, G. W. Barton, and J. D. Perkins. Robust parameter estimation in on-line optimization — part i. methodology and simulated case study. *Comput. Chem. Eng.*, 16:545–562, 1992.

[60] S. Krishnan, G. W. Barton, and J. D. Perkins. Robust parameter estimation in on-line optimization — part ii. application to an industrial process. *Comput. Chem. Eng.*, 17:663–669, 1993.

[61] P. R. Latour. Online computer optimization 1: What it is and where to do it. *Hydrocarbon Process.*, 58(6):73–82, 1979.

[62] P. R. Latour. Online computer optimization 2: Benefits and implementation. *Hydrocarbon Process.*, 58(7):219–223, 1979.

[63] R. J. Lawrence. Data reconciliation: Getting better information. *Hydrocarbon Process.*, 68(6):55–60, 1989.

[64] P. L. Lee and G. R. Sullivan. Generic model control (GMC). *Comput. Chem. Eng.*, 12:573–580, 1988.

[65] S. Lee. Simulation and control of chemical processes under generic model control utilizing the SPEEDUP simulation package. Master's thesis, University of Waterloo, 1991.

[66] M. J. Liebman, T. F. Edgar, and L. S. Lasdon. Efficient reconciliation and estimation for dynamic processes using nonlinear programming techniques. *Comput. Chem. Eng.*, 16:963–986, 1992.

[67] Y. A. Liu, H. A. McGee, Jr., and W. R. Epperly, editors. *Recent Developments in Chemical Process and Plant Design*. John Wiley and Sons, New York, 1987.

[68] R. J. Lojek and B. D. Whitehead. Integrated advanced control and online optimization in olefins plant. *Comput. Chem. Eng.*, 13:1219–1297, 1989.

[69] A. Lucia and J. Xu. Chemical process optimization using newton-like methods. *Comput. Chem. Eng.*, 14:119–138, 1990.

[70] S. Macchietto, G. Stuart, T. A. Perris, and G. R. Dissinger. Monitoring and on-line optimization of process using SPEEDUP. *Comput. Chem. Eng.*, 13:571–576, 1989.

[71] R. J. MacDonald and C. S. Howat. Data reconciliation and parameter estimation in plant performance analysis. *AIChE J.*, 34:1–8, 1988.

[72] R. S. Mah, G. M. Stanley, and D. M. Downing. Reconciliation and rectification of process flow and inventory data. *Ind. Eng. Chem. Process Des. Dev.*, 15:175–183, 1976.

[73] R. S. H. Mah. *Chemical Process Structures and Information Flows.* Butterworths, Boston, 1990.

[74] R. S. H. Mah and A. C. Tamhane. Detection of gross errors in process data. *AIChE J.*, 28:828–830, 1982.

[75] P. J. McLellan. A nonlinear refinery model for scheduling and economic evaluation. Master's thesis, University of Waterloo, 1983.

[76] M. Meyer, B. Koehret, and M. Enjalbert. Data reconciliation on multicomponent network process. *Comput. Chem. Eng.*, 17:807–817, 1993.

[77] B. A. Murtagh and M. A. Saunders. A projected lagrangian algorithm and its implementation for sparse nonlinear constraints. *Math. Program. Study*, 16:84–117, 1982.

[78] S. Narasimhan. Maximum power tests for gross error detection using likelihood ratios. *AIChE J.*, 36:1589–1591, 1990.

[79] S. Narasimhan and P. Harikumar. A method to incorporate bounds in data reconciliation and gross error detection - i. the bounded data reconciliation problem. *Comput. Chem. Eng.*, 17:1115–1120, 1993.

[80] S. Narasimhan, C. S. Kao, and R. S. H. Mah. Detecting changes of steady states using the mathematical theory of evidence. *AIChE J.*, 33:1930–1932, 1987.

[81] S. Narasimhan and R. S. H. Mah. Generalized likelihood ratio method for gross error identification. *AIChE J.*, 33:1514–1521, 1987.

[82] S. Narasimhan, R. S. H. Mah, A. C. Tamhane, J. W. Woodward, and J. C. Hale. A composite statistical test for detecting changes of steady states. *AIChE J.*, 32:1409–1418, 1986.

[83] M. R. Naysmith and P. L. Douglas. Review of real time optimization in the chemical process industries. *Dev. Chem. Eng. Min. Process.*, 3(2):67–87, 1995.

[84] The Numerical Algorithms Group Limited, Oxford. *The NAG Fortran Library Manual — Volume 9*, Mark 15 edition, June 1991.

[85] R. H. Perry and D. W. Green, editors. *Perry's Chemical Engineers' Handbook*. McGraw-Hill, New York, sixth edition, 1984.

[86] J. Picaseno-Gamiz. *A Systematic Approach for Scheduling Production in Continuous Processing Systems*. PhD thesis, University of Waterloo, 1989.

[87] M. B. Piccolo. The use of steady state simulation packages in real time optimization application. Master's thesis, University of Waterloo, 1995.

[88] J. L. Robertson. The ideal process simulator. *Chem. Eng. Prog.*, 85(10):62–66, 1989.

[89] D. K. Rollins and S. Devanathan. Unbiased estimation in dynamic data reconciliation. *AIChE J.*, 39:1330–1334, 1993.

[90] R. Rouhani and R. K. Mehra. Model algorithmic control (MAC): basic theoretical properties. *Automatica*, 18:401–414, 1982.

[91] R. W. Serth and W. A. Heenan. Gross error detection and data reconciliation in steam-metering systems. *AIChE J.*, 32:733–742, 1986.

[92] R. W. Serth, M. T. Weart, and W. A. Heenan. On the choice of regression variables in nonlinear data reconciliation. *Chem. Eng. Commun.*, 79:141–152, 1989.

[93] C. F. Shewchuk and W. Morton. The evolution of an on-line model-based optimization system. *Pulp Paper Canada*, 95(6):29–34, 1994.

[94] Simulation Sciences Inc. The Simulator. Houston, Texas, 1st Quarter 1997.

[95] G. R. Stephenson and C. F. Shewchuk. Reconciliation of process data with process simulation. *AIChE J.*, 32:247–254, 1986.

[96] W. E. Stewart, M. Caracotsios, and J. P. Sørensen. Parameter estimation from multiresponse data. *AIChE J.*, 38:641–650, 1992.

[97] W. C. Strand, O. Ferritsius, and A. V. Mokvist. Use of simulation models in the on-line control and optimization of the refining process. *Tappi J.*, 74(11):103–108, 1991.

[98] H. Takiyama, Y. Naka, and E. O'Shima. Sensor-based data reconciliation method and application to the pilot plant. *J. Chem. Eng. Japan*, 24:339–346, 1991.

[99] A. C. Tamhane and R. S. H. Mah. Data reconciliation and gross error detection in chemical process networks. *Technometrics*, 27:409–422, 1985.

[100] P. A. Terry and D. M. Himmelblau. Data rectification and gross error detection in a steady-state process via artificial neural networks. *Ind. Eng. Chem. Res.*, 32:3020–3028, 1993.

[101] I. B. Tjoa and L. T. Biegler. Simultaneous strategies for data reconciliation and gross error detection of nonlinear systems. *Comput. Chem. Eng.*, 15:679–690, 1991.

[102] E. H. Tsang and M. D. Meixell. Large scale ammonia process optimization in real-time. In *Proceedings of the 1988 American Control Conference*, volume 3, pages 1971–1974, Green Valley, Arizona, 1988. American Automatic Control Council.

[103] M. C. Wellons, A. V. Sapre, A. I. Chang, and T. L. Laird. On-line power plant optimization improves texas refiner's bottom line. *Oil Gas J.*, 92(20):53–58, 1994.

[104] P. Winter. Computer-aided process engineering: The evolution continues. *Chem. Eng. Prog.*, 88(2):76–83, 1992.

[105] Z. Zhang, R. W. Pike, and T. A. Hertwig. An approach to on-line optimization of chemical plants. *Comput. Chem. Eng.*, 19 Suppl.:S305–S310, 1995.

NOTE TO USERS

Page(s) missing in number only; text follows. Microfilmed as received.

188

UMI

Appendix A

Real Time Optimizer Run Data

This appendix contains tables summarising the real time optimizer runs for both ASPEN PLUS and SPEEDUP. The results presented in these tables are analysed in Chapter 6. Unless otherwise stated, all ASPEN PLUS real time optimizer runs were performed using version 9.2.

Table A.1: Base case ASPEN PLUS real time optimizer run.

Cycle	1	2	3	4
Economic Optimization:				
Profit ($/hr)	-9.84	169.55	215.21	247.03
Optimization iterations	10	4	12	10
Total product values ($/hr)	23035.02	23188.05	23273.30	23265.99
Total feed cost ($/hr)	22928.41	22907.03	22940.42	22897.70
Total utility costs ($/hr)	116.45	111.48	117.67	121.27
Economic Optimization Set Points:				
C1 reflux flow (bbl/day)	6913.71	7447.16	8040.20	8631.66
C1 tray 3 temperature (°F)	181.26	173.25	166.41	161.44
DIST-C4 C_5 mole fraction	0.11882	0.06883	0.03562	0.01633
C1 reboiler temperature (°F)	595.10	602.16	609.97	612.36
C2 reflux flow (bbl/day)	13951.69	13682.32	13610.05	13556.18
C4 feed flow (bbl/day)	5998.29	6279.27	6396.15	6638.08
C5 feed flow (bbl/day)	3334.48	3452.69	3341.47	3219.25
C2 reboiler temperature (°F)	595.10	602.16	609.97	612.36
HV-ISO 10% TBP temperature (°F)	197.38	199.39	194.49	198.87
HV-ISO 90% TBP temperature (°F)	351.62	364.27	371.09	377.21
LT-DIST 10% TBP temperature (°F)	338.26	357.10	364.90	370.57
LT-DIST 90% TBP temperature (°F)	496.25	553.97	554.04	553.90
Economic Optimization Inequality Constraints:				
C1 reflux flow (bbl/day)	6913.71	7447.16	8040.20	8631.66
C2 reflux flow (bbl/day)	13951.69	13682.32	13610.05	13556.18
C4 feed flow (bbl/day)	5998.29	6279.27	6396.15	6638.08
C5 feed flow (bbl/day)	3334.48	3452.69	3341.47	3219.25
C1 tray 3 temperature (°F)	181.26	173.25	166.41	161.44
C1 reboiler temperature (°F)	595.10	602.16	609.97	612.36
C2 tray 2 temperature (°F)	153.02	154.64	155.96	157.11
C2 reboiler temperature (°F)	651.22	642.93	636.87	630.89
DIST-C4 C_5 mole fraction	0.11882	0.06883	0.03562	0.01633
LT-ISO Reid vapour pressure (psi)	14.000[a]	12.395	11.613	11.410
LT-ISO bubble point temperature (°F)	115.05	121.52	124.89	125.88
HV-ISO 10% TBP temperature (°F)	197.38	199.39	194.49	198.87
HV-ISO 90% TBP temperature (°F)	351.62	364.27	371.09	377.21
LT-DIST 10% TBP temperature (°F)	338.26	357.10	364.90	370.57
LT-DIST 90% TBP temperature (°F)	496.25	553.97[a]	554.04[a]	553.90[a]

continued on the next page ...

[a]constraint upper bound active

... Table A.1 continued from the previous page

Cycle	1	2	3	4
Product Standard Volume Flows After Economic Optimization:				
Vapour Product (bbl/day)	1024.90	998.71	974.53	945.39
C_4-Distillate Product (bbl/day)	902.05	913.00	922.92	927.22
Light Iso Product (bbl/day)	1559.61	1600.17	1659.19	1731.70
Heavy Iso Product (bbl/day)	4202.01	4447.05	4551.61	4581.92
Light Distillate Product (bbl/day)	1952.44	2399.79	2297.54	2181.41
Splitter Bottoms Product (bbl/day)	12218.00	11480.00	11464.53	11462.33
Reboiler Heat Duties After Economic Optimization:				
Stabilizer (C1) Reboiler (MMBTU/hr)	21.2420	23.4262	26.3919	28.5314
Splitter (C2) Reboiler (MMBTU/hr)	18.0814	14.2192	13.3428	12.4193
Economic Optimization Marginal Stream Values:				
Light Iso Marginal Stream Value ($/m^3)	182.78	184.77	185.74	185.99
Data Reconciliation:				
Final objective function value	39.732	11.142	11.508	11.731
Initial objective function value	255.659	31.391	31.391	31.393
Data reconciliation iterations	15	21	19	22

Table A.2: Base case SPEEDUP real time optimizer run.

Cycle	1	2
Economic Optimization:		
Profit ($/hr)	277.36	379.52
Optimization iterations	33	23
Objective function weight	1.2	1.2
Total product values ($/hr)	21848.83	22052.22
Total feed cost ($/hr)	21456.45	21557.89
Total utility costs ($/hr)	115.01	114.81
Economic Optimization Set Points:		
C1 reflux flow (bbl/day)	6634.96	7143.64
C1 tray 3 temperature (°F)	154.70	155.47
DIST-C4 C_5 mole fraction	0.00676	0.00500
C1 reboiler temperature (°F)	566.74	582.34
C2 reflux flow (bbl/day)	12389.58	11410.13
C4 feed flow (bbl/day)	5418.61	5600.68
C5 feed flow (bbl/day)	3504.00	3663.28
C2 reboiler temperature (°F)	566.74	582.34
HV-ISO 10% TBP temperature (°F)	210.96	233.85
HV-ISO 90% TBP temperature (°F)	362.14	384.48
LT-DIST 10% TBP temperature (°F)	346.63	372.08
LT-DIST 90% TBP temperature (°F)	498.93	554.00
Economic Optimization Inequality Constraints:		
C1 reflux flow (bbl/day)	6634.96	7143.64
C2 reflux flow (bbl/day)	12389.58	11410.13
C4 feed flow (bbl/day)	5418.61	5600.68
C5 feed flow (bbl/day)	3504.00	3663.28
C1 tray 3 temperature (°F)	154.70	155.47
C1 reboiler temperature (°F)	566.74	582.34
C2 tray 2 temperature (°F)	167.80	185.46
C2 reboiler temperature (°F)	668.35	667.03
DIST-C4 C_5 mole fraction	0.00676	0.00500[a]
LT-ISO Reid vapour pressure (psi)	14.000[b]	10.283
LT-ISO bubble point temperature (°F)	115.54	133.10
HV-ISO 10% TBP temperature (°F)	210.96	233.85
HV-ISO 90% TBP temperature (°F)	362.14	384.48
LT-DIST 10% TBP temperature (°F)	346.63	372.08
LT-DIST 90% TBP temperature (°F)	498.93	554.00[b]

continued on the next page ...

[a]constraint lower bound active
[b]constraint upper bound active

... Table A.2 continued from the previous page

Cycle	1	2
Product Standard Volume Flows After Economic Optimization:		
Vapour Product (bbl/day)	4785.44	4808.07
C_4-Distillate Product (bbl/day)	192.33	201.76
Light Iso Product (bbl/day)	185.05	192.66
Heavy Iso Product (bbl/day)	493.36	582.19
Light Distillate Product (bbl/day)	884.07	878.68
Splitter Bottoms Product (bbl/day)	437.75	484.33
Reboiler Heat Duties After Economic Optimization:		
Stabilizer (C1) Reboiler (MMBTU/hr)	18.3094	21.0414
Splitter (C2) Reboiler (MMBTU/hr)	20.5302	17.7294
Economic Optimization Marginal Stream Values:		
Light Iso Marginal Stream Value ($/m^3)	182.78	187.39
Data Reconciliation:		
Final objective function value	67.518	20.531
Initial objective function value	257.083	33.225
Data reconciliation iterations	33	35
Data reconciliation weights	1.2	1

Table A.3: Base case with no measurement noise ASPEN PLUS real time optimizer run.

Cycle	1	2	3	4
Economic Optimization:				
Profit ($/hr)	-9.84	183.96	256.28	319.91
Optimization iterations	10	7	12	7
Total product values ($/hr)	23035.02	23224.68	23298.84	23362.20
Total feed cost ($/hr)	22928.41	22928.41	22928.41	22928.41
Total utility costs ($/hr)	116.45	112.31	114.15	113.88
Economic Optimization Set Points:				
C1 reflux flow (bbl/day)	6913.71	6814.09	6695.27	6603.68
C1 tray 3 temperature (°F)	181.26	180.16	180.91	179.97
DIST-C4 C_5 mole fraction	0.11882	0.11187	0.11657	0.11071
C1 reboiler temperature (°F)	595.10	590.70	594.91	590.20
C2 reflux flow (bbl/day)	13951.69	13779.72	13610.48	13642.27
C4 feed flow (bbl/day)	5998.29	6253.26	6396.92	6454.85
C5 feed flow (bbl/day)	3334.48	3732.64	3598.05	3469.71
C2 reboiler temperature (°F)	595.10	590.70	594.91	590.20
HV-ISO 10% TBP temperature (°F)	197.38	192.75	202.13	199.85
HV-ISO 90% TBP temperature (°F)	351.62	360.69	373.88	379.42
LT-DIST 10% TBP temperature (°F)	338.26	352.10	367.18	370.39
LT-DIST 90% TBP temperature (°F)	496.25	554.03	553.87	553.82
Economic Optimization Inequality Constraints:				
C1 reflux flow (bbl/day)	6913.71	6814.09	6695.27	6603.68
C2 reflux flow (bbl/day)	13951.69	13779.72	13610.48	13642.27
C4 feed flow (bbl/day)	5998.29	6253.26	6396.92	6454.85
C5 feed flow (bbl/day)	3334.48	3732.64	3598.05	3469.71
C1 tray 3 temperature (°F)	181.26	180.16	180.91	179.97
C1 reboiler temperature (°F)	595.10	590.70	594.91	590.20
C2 tray 2 temperature (°F)	153.02	154.61	157.98	156.17
C2 reboiler temperature (°F)	651.22	656.22	661.14	665.79
DIST-C4 C_5 mole fraction	0.11882	0.11187	0.11657	0.11071
LT-ISO Reid vapour pressure (psi)	14.000[a]	14.000[a]	13.412	13.980[a]
LT-ISO bubble point temperature (°F)	115.05	115.18	117.92	115.42
HV-ISO 10% TBP temperature (°F)	197.38	192.75	202.13	199.85
HV-ISO 90% TBP temperature (°F)	351.62	360.69	373.88	379.42
LT-DIST 10% TBP temperature (°F)	338.26	352.10	367.18	370.39
LT-DIST 90% TBP temperature (°F)	496.25	554.03[a]	553.87[a]	553.82[a]

continued on the next page ...

[a]constraint upper bound active

... Table A.3 continued from the previous page

Cycle	1	2	3	4
Product Standard Volume Flows After Economic Optimization:				
Vapour Product (bbl/day)	1024.90	959.01	910.53	853.57
C_4-Distillate Product (bbl/day)	902.05	952.64	1014.12	1056.83
Light Iso Product (bbl/day)	1559.61	1637.45	1724.54	1691.79
Heavy Iso Product (bbl/day)	4202.01	4393.49	4640.23	4816.03
Light Distillate Product (bbl/day)	1952.44	2514.66	2279.34	2229.20
Splitter Bottoms Product (bbl/day)	12218.00	11401.62	11291.19	11212.03
Reboiler Heat Duties After Economic Optimization:				
Stabilizer (C1) Reboiler (MMBTU/hr)	21.2420	20.9219	20.9620	20.6658
Splitter (C2) Reboiler (MMBTU/hr)	18.0814	17.0053	17.5860	17.7914
Economic Optimization Marginal Stream Values:				
Light Iso Marginal Stream Value ($/m^3)	182.78	182.77	183.50	182.80

continued on the next page ...

... Table A.3 continued from the previous page

Cycle	5	6	7	8
Economic Optimization:				
Profit ($/hr)	393.13	435.69	495.55	553.97
Optimization iterations	5	10	6	10
Total product values ($/hr)	23434.55	23475.25	23536.29	23598.37
Total feed cost ($/hr)	22928.41	22928.41	22928.41	22928.41
Total utility costs ($/hr)	113.01	111.15	112.34	115.98
Economic Optimization Set Points:				
C1 reflux flow (bbl/day)	6504.24	6428.55	6359.33	6278.80
C1 tray 3 temperature (°F)	179.75	178.34	176.79	176.23
DIST-C4 C_5 mole fraction	0.10933	0.10103	0.09249	0.08949
C1 reboiler temperature (°F)	592.24	592.71	587.74	583.68
C2 reflux flow (bbl/day)	13567.97	13358.32	13087.92	12841.07
C4 feed flow (bbl/day)	6526.93	6253.07	6095.12	6105.22
C5 feed flow (bbl/day)	3349.02	3503.47	3684.20	3528.07
C2 reboiler temperature (°F)	592.24	592.71	587.74	583.68
HV-ISO 10% TBP temperature (°F)	204.03	206.50	208.80	213.20
HV-ISO 90% TBP temperature (°F)	387.67	383.33	381.43	389.02
LT-DIST 10% TBP temperature (°F)	375.11	368.37	362.46	366.61
LT-DIST 90% TBP temperature (°F)	557.76	554.01	554.01	554.04
Economic Optimization Inequality Constraints:				
C1 reflux flow (bbl/day)	6504.24	6428.55	6359.33	6278.80
C2 reflux flow (bbl/day)	13567.97	13358.32	13087.92	12841.07
C4 feed flow (bbl/day)	6526.93	6253.07	6095.12	6105.22
C5 feed flow (bbl/day)	3349.02	3503.47	3684.20	3528.07
C1 tray 3 temperature (°F)	179.75	178.34	176.79	176.23
C1 reboiler temperature (°F)	592.24	592.71	587.74	583.68
C2 tray 2 temperature (°F)	157.18	159.11	162.46	166.43
C2 reboiler temperature (°F)	670.79	674.19	679.19	684.17
DIST-C4 C_5 mole fraction	0.10933	0.10103	0.09249	0.08949
LT-ISO Reid vapour pressure (psi)	13.946[a]	13.997[a]	14.000[a]	13.790
LT-ISO bubble point temperature (°F)	115.66	115.52	115.60	116.61
HV-ISO 10% TBP temperature (°F)	204.03	206.50	208.80	213.20
HV-ISO 90% TBP temperature (°F)	387.67	383.33	381.43	389.02
LT-DIST 10% TBP temperature (°F)	375.11	368.37	362.46	366.61
LT-DIST 90% TBP temperature (°F)	557.76[a]	554.01[a]	554.01[a]	554.04[a]

continued on the next page ...

[a]constraint upper bound active

... Table A.3 continued from the previous page

Cycle	5	6	7	8
Product Standard Volume Flows After Economic Optimization:				
Vapour Product (bbl/day)	803.57	749.13	697.78	654.75
C_4-Distillate Product (bbl/day)	1103.76	1136.59	1165.10	1199.40
Light Iso Product (bbl/day)	1732.16	1818.47	1912.97	1999.43
Heavy Iso Product (bbl/day)	4964.27	4742.21	4582.53	4694.40
Light Distillate Product (bbl/day)	2204.96	2445.40	2722.57	2630.26
Splitter Bottoms Product (bbl/day)	11050.43	10967.38	10778.12	10680.81
Reboiler Heat Duties After Economic Optimization:				
Stabilizer (C1) Reboiler (MMBTU/hr)	20.5113	20.1392	19.7603	19.5556
Splitter (C2) Reboiler (MMBTU/hr)	17.6509	17.3952	18.1745	19.6115
Economic Optimization Marginal Stream Values:				
Light Iso Marginal Stream Value ($/m^3)	182.84	182.78	182.78	183.04

continued on the next page ...

... Table A.3 continued from the previous page

Cycle	9	10
Economic Optimization:		
Profit ($/hr)	602.92	654.81
Optimization iterations	6	10
Total product values ($/hr)	23648.49	23703.22
Total feed cost ($/hr)	22928.41	22928.41
Total utility costs ($/hr)	117.16	120.00
Economic Optimization Set Points:		
C1 reflux flow (bbl/day)	6235.53	6190.11
C1 tray 3 temperature (°F)	173.54	172.83
DIST-C4 C_5 mole fraction	0.07630	0.07304
C1 reboiler temperature (°F)	581.63	578.51
C2 reflux flow (bbl/day)	12598.46	12496.08
C4 feed flow (bbl/day)	6055.62	6208.55
C5 feed flow (bbl/day)	3693.02	3536.63
C2 reboiler temperature (°F)	581.63	578.51
HV-ISO 10% TBP temperature (°F)	215.82	219.00
HV-ISO 90% TBP temperature (°F)	387.90	395.04
LT-DIST 10% TBP temperature (°F)	365.03	374.28
LT-DIST 90% TBP temperature (°F)	554.00	553.99
Economic Optimization Inequality Constraints:		
C1 reflux flow (bbl/day)	6235.53	6190.11
C2 reflux flow (bbl/day)	12598.46	12496.08
C4 feed flow (bbl/day)	6055.62	6208.55
C5 feed flow (bbl/day)	3693.02	3536.63
C1 tray 3 temperature (°F)	173.54	172.83
C1 reboiler temperature (°F)	581.63	578.51
C2 tray 2 temperature (°F)	168.44	169.63
C2 reboiler temperature (°F)	689.17	694.17
DIST-C4 C_5 mole fraction	0.07630	0.07304
LT-ISO Reid vapour pressure (psi)	13.985[a]	14.005[a]
LT-ISO bubble point temperature (°F)	115.79	115.72
HV-ISO 10% TBP temperature (°F)	215.82	219.00
HV-ISO 90% TBP temperature (°F)	387.90	395.04
LT-DIST 10% TBP temperature (°F)	365.03	374.28
LT-DIST 90% TBP temperature (°F)	554.00[a]	553.99[a]

continued on the next page ...

[a]constraint upper bound active

... Table A.3 continued from the previous page

Cycle	9	10
Product Standard Volume Flows After Economic Optimization:		
Vapour Product (bbl/day)	605.32	578.12
C_4-Distillate Product (bbl/day)	1212.77	1230.23
Light Iso Product (bbl/day)	2100.53	2140.28
Heavy Iso Product (bbl/day)	4575.44	4835.56
Light Distillate Product (bbl/day)	2752.47	2522.88
Splitter Bottoms Product (bbl/day)	10612.88	10551.73
Reboiler Heat Duties After Economic Optimization:		
Stabilizer (C1) Reboiler (MMBTU/hr)	19.0359	18.8585
Splitter (C2) Reboiler (MMBTU/hr)	20.5287	21.6630
Economic Optimization Marginal Stream Values:		
Light Iso Marginal Stream Value ($/m^3)	182.79	182.77

Table A.4: Base case with no measurement noise SPEEDUP real time optimizer run.

Cycle	1	2	3
Economic Optimization:			
Profit ($/hr)	277.36	430.27	525.25
Optimization iterations	33	18	18
Objective function weight	1.2	1.99	1.99
Total product values ($/hr)	21848.83	21978.67	22079.88
Total feed cost ($/hr)	21456.45	21440.37	21441.27
Total utility costs ($/hr)	115.01	108.03	113.36
Economic Optimization Set Points:			
C1 reflux flow (bbl/day)	6634.96	6634.33	6663.53
C1 tray 3 temperature (°F)	154.70	153.73	153.67
DIST-C4 C_5 mole fraction	0.00676	0.00500	0.00500
C1 reboiler temperature (°F)	566.74	566.34	568.20
C2 reflux flow (bbl/day)	12389.58	11245.98	11355.81
C4 feed flow (bbl/day)	5418.61	5558.39	5365.57
C5 feed flow (bbl/day)	3504.00	3533.19	3427.19
C2 reboiler temperature (°F)	566.74	566.34	568.20
HV-ISO 10% TBP temperature (°F)	210.96	234.32	235.31
HV-ISO 90% TBP temperature (°F)	362.14	385.13	393.08
LT-DIST 10% TBP temperature (°F)	346.63	373.22	376.95
LT-DIST 90% TBP temperature (°F)	498.93	554.00	554.00
Economic Optimization Inequality Constraints:			
C1 reflux flow (bbl/day)	6634.96	6634.33	6663.53
C2 reflux flow (bbl/day)	12389.58	11245.98	11355.81
C4 feed flow (bbl/day)	5418.61	5558.39	5365.57
C5 feed flow (bbl/day)	3504.00	3533.19	3427.19
C1 tray 3 temperature (°F)	154.70	153.73	153.67
C1 reboiler temperature (°F)	566.74	566.34	568.20
C2 tray 2 temperature (°F)	167.80	184.28	185.70
C2 reboiler temperature (°F)	668.35	665.62	679.97
DIST-C4 C_5 mole fraction	0.00676	0.00500[a]	0.00500[a]
LT-ISO Reid vapour pressure (psi)	14.000[b]	12.080	11.784
LT-ISO bubble point temperature (°F)	115.54	124.64	126.03
HV-ISO 10% TBP temperature (°F)	210.96	234.32	235.31
HV-ISO 90% TBP temperature (°F)	362.14	385.13	393.08
LT-DIST 10% TBP temperature (°F)	346.63	373.22	376.95
LT-DIST 90% TBP temperature (°F)	498.93	554.00[b]	554.00[b]

continued on the next page ...

[a]constraint lower bound active
[b]constraint upper bound active

... Table A.4 continued from the previous page

Cycle	1	2	3
Product Standard Volume Flows After Economic Optimization:			
Vapour Product (bbl/day)	4785.44	4781.86	4782.06
C_4-Distillate Product (bbl/day)	192.33	182.18	173.45
Light Iso Product (bbl/day)	185.05	194.71	206.50
Heavy Iso Product (bbl/day)	493.36	600.62	601.15
Light Distillate Product (bbl/day)	884.07	874.42	928.14
Splitter Bottoms Product (bbl/day)	437.75	449.78	476.21
Reboiler Heat Duties After Economic Optimization:			
Stabilizer (C1) Reboiler (MMBTU/hr)	18.3094	18.2981	18.5659
Splitter (C2) Reboiler (MMBTU/hr)	20.5302	18.1831	19.7151
Economic Optimization Marginal Stream Values:			
Light Iso Marginal Stream Value ($/m^3)	182.78	185.16	185.53

continued on the next page ...

... Table A.4 continued from the previous page

Cycle	4	5
Economic Optimization:		
Profit ($/hr)	619.08	698.65
Optimization iterations	24	34
Objective function weight	1.99	1.49
Total product values ($/hr)	22184.17	22265.85
Total feed cost ($/hr)	21445.52	21445.52
Total utility costs ($/hr)	119.58	121.68
Economic Optimization Set Points:		
C1 reflux flow (bbl/day)	6698.10	6586.74
C1 tray 3 temperature (°F)	153.61	153.19
DIST-C4 C_5 mole fraction	0.00500	0.00500
C1 reboiler temperature (°F)	570.19	569.12
C2 reflux flow (bbl/day)	11549.58	11945.53
C4 feed flow (bbl/day)	5181.86	4979.68
C5 feed flow (bbl/day)	3325.22	3213.04
C2 reboiler temperature (°F)	570.19	569.12
HV-ISO 10% TBP temperature (°F)	235.72	234.06
HV-ISO 90% TBP temperature (°F)	400.70	410.38
LT-DIST 10% TBP temperature (°F)	380.79	382.21
LT-DIST 90% TBP temperature (°F)	554.00	554.00
Economic Optimization Inequality Constraints:		
C1 reflux flow (bbl/day)	6698.10	6586.74
C2 reflux flow (bbl/day)	11549.58	11945.53
C4 feed flow (bbl/day)	5181.86	4979.68
C5 feed flow (bbl/day)	3325.22	3213.04
C1 tray 3 temperature (°F)	153.61	153.19
C1 reboiler temperature (°F)	570.19	569.12
C2 tray 2 temperature (°F)	186.46	183.83
C2 reboiler temperature (°F)	693.30	700.00
DIST-C4 C_5 mole fraction	0.00500[a]	0.00500[a]
LT-ISO Reid vapour pressure (psi)	11.536	11.919
LT-ISO bubble point temperature (°F)	127.18	125.26
HV-ISO 10% TBP temperature (°F)	235.72	234.06
HV-ISO 90% TBP temperature (°F)	400.70	410.38
LT-DIST 10% TBP temperature (°F)	380.79	382.21
LT-DIST 90% TBP temperature (°F)	554.00[b]	554.00[b]

continued on the next page ...

[a]constraint lower bound active
[b]constraint upper bound active

... Table A.4 continued from the previous page

Cycle	4	5
Product Standard Volume Flows After Economic Optimization:		
Vapour Product (bbl/day)	4783.01	4783.01
C_4-Distillate Product (bbl/day)	164.20	154.29
Light Iso Product (bbl/day)	219.20	229.33
Heavy Iso Product (bbl/day)	596.79	579.04
Light Distillate Product (bbl/day)	983.67	1035.38
Splitter Bottoms Product (bbl/day)	504.52	531.99
Reboiler Heat Duties After Economic Optimization:		
Stabilizer (C1) Reboiler (MMBTU/hr)	18.8524	18.6900
Splitter (C2) Reboiler (MMBTU/hr)	21.5276	22.3992
Economic Optimization Marginal Stream Values:		
Light Iso Marginal Stream Value ($\$/m^3$)	185.83	185.36

Table A.5: ASPEN PLUS real time optimizer run with Heavy Iso price at $\$173.89/m^3$.

Cycle	1	2	3	4
Economic Optimization:				
Profit ($/hr)	-565.41	-381.52	-382.71	-365.11
Optimization iterations	5	7	6	11
Total product values ($/hr)	22480.47	22645.96	22695.47	22751.31
Total feed cost ($/hr)	22928.41	22914.01	22957.44	22989.30
Total utility costs ($/hr)	117.47	113.47	120.74	127.13
Economic Optimization Set Points:				
C1 reflux flow (bbl/day)	6913.45	7489.73	8090.05	8684.70
C1 tray 3 temperature (°F)	181.30	171.59	163.80	158.02
DIST-C4 C_5 mole fraction	0.11912	0.06024	0.02584	0.00587
C1 reboiler temperature (°F)	595.27	606.28	612.59	617.62
C2 reflux flow (bbl/day)	13944.95	13854.65	13029.30	12359.16
C4 feed flow (bbl/day)	5590.84	5847.03	5650.14	5404.42
C5 feed flow (bbl/day)	3551.71	4006.10	4052.13	4134.72
C2 reboiler temperature (°F)	595.27	606.28	612.59	617.62
HV-ISO 10% TBP temperature (°F)	195.84	198.34	196.86	197.22
HV-ISO 90% TBP temperature (°F)	336.66	348.00	336.62	328.19
LT-DIST 10% TBP temperature (°F)	316.29	329.26	320.55	308.87
LT-DIST 90% TBP temperature (°F)	482.14	554.04	553.06	554.17
Economic Optimization Inequality Constraints:				
C1 reflux flow (bbl/day)	6913.45	7489.73	8090.05	8684.70
C2 reflux flow (bbl/day)	13944.95	13854.65	13029.30	12359.16
C4 feed flow (bbl/day)	5590.84	5847.03	5650.14	5404.42
C5 feed flow (bbl/day)	3551.71	4006.10	4052.13	4134.72
C1 tray 3 temperature (°F)	181.30	171.59	163.80	158.02
C1 reboiler temperature (°F)	595.27	606.28	612.59	617.62
C2 tray 2 temperature (°F)	152.95	154.83	154.09	154.10
C2 reboiler temperature (°F)	651.22	636.93	620.82	610.89
DIST-C4 C_5 mole fraction	0.11912	0.06024	0.02584	0.00587
LT-ISO Reid vapour pressure (psi)	14.000[a]	11.992	11.825	11.827
LT-ISO bubble point temperature (°F)	115.05	123.21	123.64	123.55
HV-ISO 10% TBP temperature (°F)	195.84	198.34	196.86	197.22
HV-ISO 90% TBP temperature (°F)	336.66	348.00	336.62	328.19
LT-DIST 10% TBP temperature (°F)	316.29	329.26	320.55	308.87
LT-DIST 90% TBP temperature (°F)	482.14	554.04[a]	553.06[a]	554.17[a]

continued on the next page ...

[a]constraint upper bound active

... Table A.5 continued from the previous page

Cycle	1	2	3	4
Product Standard Volume Flows After Economic Optimization:				
Vapour Product (bbl/day)	1025.49	999.33	969.33	945.92
C_4-Distillate Product (bbl/day)	902.17	916.97	919.11	921.11
Light Iso Product (bbl/day)	1558.48	1595.21	1621.45	1662.65
Heavy Iso Product (bbl/day)	3740.08	3804.51	3524.96	3237.90
Light Distillate Product (bbl/day)	2297.20	3116.47	3090.40	3243.15
Splitter Bottoms Product (bbl/day)	12335.57	11411.69	11761.48	11906.34
Reboiler Heat Duties After Economic Optimization:				
Stabilizer (C1) Reboiler (MMBTU/hr)	21.2511	24.6103	27.7607	30.4623
Splitter (C2) Reboiler (MMBTU/hr)	18.4187	13.7067	13.0121	12.4678
Economic Optimization Marginal Stream Values:				
Light Iso Marginal Stream Value ($/m^3)	182.77	185.27	185.47	185.47
Data Reconciliation:				
Final objective function value	39.732	11.891	15.020	13.885
Initial objective function value	255.659	31.394	31.393	31.388
Data reconciliation iterations	15	18	20	21

Table A.6: ASPEN PLUS real time optimizer run with Light Distillate price at $\$200.52/m^3$.

Cycle	1	2	3	4
Economic Optimization:				
Profit ($/hr)	239.54	580.75	567.85	545.78
Optimization iterations	8	7	8	3
Total product values ($/hr)	23285.39	23599.77	23594.08	23641.26
Total feed cost ($/hr)	22928.41	22901.51	22905.40	22967.47
Total utility costs ($/hr)	117.44	117.51	120.83	128.02
Economic Optimization Set Points:				
C1 reflux flow (bbl/day)	6913.39	7439.98	8046.32	8650.61
C1 tray 3 temperature (°F)	181.28	172.98	166.60	159.42
DIST-C4 C_5 mole fraction	0.11899	0.06710	0.03623	0.01015
C1 reboiler temperature (°F)	595.14	605.12	607.09	609.89
C2 reflux flow (bbl/day)	13958.13	14185.04	13734.76	13136.78
C4 feed flow (bbl/day)	5585.31	6032.91	5765.59	5560.68
C5 feed flow (bbl/day)	3586.48	4177.17	4233.95	4115.21
C2 reboiler temperature (°F)	595.14	605.12	607.09	609.89
HV-ISO 10% TBP temperature (°F)	195.84	189.44	177.87	176.56
HV-ISO 90% TBP temperature (°F)	336.09	347.28	332.51	328.29
LT-DIST 10% TBP temperature (°F)	315.69	328.09	312.97	306.72
LT-DIST 90% TBP temperature (°F)	482.72	553.98	553.98	553.98
Economic Optimization Inequality Constraints:				
C1 reflux flow (bbl/day)	6913.39	7439.98	8046.32	8650.61
C2 reflux flow (bbl/day)	13958.13	14185.04	13734.76	13136.78
C4 feed flow (bbl/day)	5585.31	6032.91	5765.59	5560.68
C5 feed flow (bbl/day)	3586.48	4177.17	4233.95	4115.21
C1 tray 3 temperature (°F)	181.28	172.98	166.60	159.42
C1 reboiler temperature (°F)	595.14	605.12	607.09	609.89
C2 tray 2 temperature (°F)	152.96	150.06	142.65	142.03
C2 reboiler temperature (°F)	651.22	649.17	631.71	621.86
DIST-C4 C_5 mole fraction	0.11899	0.06710	0.03623	0.01015
LT-ISO Reid vapour pressure (psi)	14.000[a]	12.645	13.236	13.304
LT-ISO bubble point temperature (°F)	115.05	119.87	116.46	115.99
HV-ISO 10% TBP temperature (°F)	195.84	189.44	177.87	176.56[b]
HV-ISO 90% TBP temperature (°F)	336.09	347.28	332.51	328.29
LT-DIST 10% TBP temperature (°F)	315.69	328.09	312.97	306.72
LT-DIST 90% TBP temperature (°F)	482.72	553.98[a]	553.98[a]	553.98[a]

continued on the next page ...

[a]constraint upper bound active
[b]constraint lower bound active

... Table A.6 continued from the previous page

Cycle	1	2	3	4
Product Standard Volume Flows After Economic Optimization:				
Vapour Product (bbl/day)	1025.12	1004.98	970.29	940.21
C_4-Distillate Product (bbl/day)	902.25	918.66	926.18	929.92
Light Iso Product (bbl/day)	1559.68	1453.38	1335.76	1368.63
Heavy Iso Product (bbl/day)	3722.62	3969.22	3728.04	3605.43
Light Distillate Product (bbl/day)	2327.59	3291.43	3439.06	3362.45
Splitter Bottoms Product (bbl/day)	12321.82	11195.91	11437.34	11689.72
Reboiler Heat Duties After Economic Optimization:				
Stabilizer (C1) Reboiler (MMBTU/hr)	21.2471	24.4809	26.6818	28.9070
Splitter (C2) Reboiler (MMBTU/hr)	18.4114	15.2027	14.1227	14.3239
Economic Optimization Marginal Stream Values:				
Light Iso Marginal Stream Value ($/m^3)	182.78	184.46	183.72	183.64
Data Reconciliation:				
Final objective function value	39.732	11.785	14.862	14.275
Initial objective function value	255.659	31.396	31.391	31.394
Data reconciliation iterations	15	21	21	18

Table A.7: ASPEN PLUS real time optimizer run with Heavy Iso price at $\$173.89/m^3$ and Light Distillate price at $\$200.52/m^3$.

Cycle	1	2	3
Economic Optimization:			
Profit ($/hr)	-253.29	59.76	121.25
Optimization iterations	8	12	14
Total product values ($/hr)	22792.61	23358.45	23434.45
Total feed cost ($/hr)	22928.41	23184.91	23196.20
Total utility costs ($/hr)	117.48	113.78	117.00
Economic Optimization Set Points:			
C1 reflux flow (bbl/day)	6913.36	7333.40	7908.27
C1 tray 3 temperature (°F)	181.28	182.09	174.22
DIST-C4 C_5 mole fraction	0.11899	0.12067	0.07179
C1 reboiler temperature (°F)	595.27	596.42	599.63
C2 reflux flow (bbl/day)	13957.85	13906.51	13408.01
C4 feed flow (bbl/day)	5578.04	5762.63	5747.07
C5 feed flow (bbl/day)	3584.66	4409.31	4591.09
C2 reboiler temperature (°F)	595.27	596.42	599.63
HV-ISO 10% TBP temperature (°F)	195.81	188.68	186.64
HV-ISO 90% TBP temperature (°F)	335.82	345.72	332.39
LT-DIST 10% TBP temperature (°F)	315.18	325.74	310.85
LT-DIST 90% TBP temperature (°F)	482.10	554.01	554.03
Economic Optimization Inequality Constraints:			
C1 reflux flow (bbl/day)	6913.36	7333.40	7908.27
C2 reflux flow (bbl/day)	13957.85	13906.51	13408.01
C4 feed flow (bbl/day)	5578.04	5762.63	5747.07
C5 feed flow (bbl/day)	3584.66	4409.31	4591.09
C1 tray 3 temperature (°F)	181.28	182.09	174.22
C1 reboiler temperature (°F)	595.27	596.42	599.63
C2 tray 2 temperature (°F)	152.96	155.80	155.81
C2 reboiler temperature (°F)	651.22	657.81	649.80
DIST-C4 C_5 mole fraction	0.11899	0.12067	0.07179
LT-ISO Reid vapour pressure (psi)	14.000[a]	13.339	12.543
LT-ISO bubble point temperature (°F)	115.05	118.00	121.09
HV-ISO 10% TBP temperature (°F)	195.81	188.68	186.64
HV-ISO 90% TBP temperature (°F)	335.82	345.72	332.39
LT-DIST 10% TBP temperature (°F)	315.18	325.74	310.85
LT-DIST 90% TBP temperature (°F)	482.10	554.01[a]	554.03[a]

continued on the next page ...

[a]constraint upper bound active

... Table A.7 continued from the previous page

Cycle	1	2	3
Product Standard Volume Flows After Economic Optimization:			
Vapour Product (bbl/day)	1025.12	1024.17	994.90
C_4-Distillate Product (bbl/day)	902.25	937.02	931.90
Light Iso Product (bbl/day)	1559.68	1639.70	1686.47
Heavy Iso Product (bbl/day)	3714.51	3778.85	3401.15
Light Distillate Product (bbl/day)	2331.66	3382.84	3567.86
Splitter Bottoms Product (bbl/day)	12325.98	11340.73	11532.20
Reboiler Heat Duties After Economic Optimization:			
Stabilizer (C1) Reboiler (MMBTU/hr)	21.2472	21.2672	22.5092
Splitter (C2) Reboiler (MMBTU/hr)	18.4250	17.1542	17.0012
Economic Optimization Marginal Stream Values:			
Light Iso Marginal Stream Value ($/m^3)	182.78	183.60	184.58
Data Reconciliation:			
Final objective function value	39.732	19.878	16.359
Initial objective function value	255.659	31.394	31.395
Data reconciliation iterations	15	6	15

Table A.8: Winter ASPEN PLUS v9.3 real time optimizer run.

Cycle	1	2	3	4
Economic Optimization:				
Profit ($/hr)	-170.90	137.13	294.02	404.23
Optimization iterations	4	5	11	10
Total product values ($/hr)	22958.75	23285.14	23365.12	23426.27
Total feed cost ($/hr)	23009.43	23028.10	22941.86	22885.32
Total utility costs ($/hr)	120.21	119.91	129.24	136.72
Economic Optimization Set Points:				
C1 reflux flow (bbl/day)	7012.25	7569.65	8154.29	8705.16
C1 tray 3 temperature (°F)	172.17	161.52	157.22	157.73
DIST-C4 C_5 mole fraction	0.06538	0.02080	0.00500	0.00513
C1 reboiler temperature (°F)	587.59	597.02	600.06	601.49
C2 reflux flow (bbl/day)	14212.28	14207.00	14166.87	14157.81
C4 feed flow (bbl/day)	5647.83	5951.76	5831.13	5815.26
C5 feed flow (bbl/day)	3586.57	4149.26	4488.58	4793.16
C2 reboiler temperature (°F)	587.59	597.02	600.06	601.49
HV-ISO 10% TBP temperature (°F)	195.23	199.55	191.67	196.92
HV-ISO 90% TBP temperature (°F)	335.84	348.39	349.79	348.03
LT-DIST 10% TBP temperature (°F)	312.59	323.86	320.67	314.69
LT-DIST 90% TBP temperature (°F)	481.35	549.80	553.93	553.96
Economic Optimization Inequality Constraints:				
C1 reflux flow (bbl/day)	7012.25	7569.65	8154.29	8705.16
C2 reflux flow (bbl/day)	14212.28	14207.00	14166.87	14157.81
C4 feed flow (bbl/day)	5647.83	5951.76	5831.13	5815.26
C5 feed flow (bbl/day)	3586.57	4149.26	4488.58	4793.16
C1 tray 3 temperature (°F)	172.17	161.52	157.22	157.73
C1 reboiler temperature (°F)	587.59	597.02	600.06	601.49
C2 tray 2 temperature (°F)	151.82	154.51	156.85	159.65
C2 reboiler temperature (°F)	652.65	651.52	660.78	665.42
DIST-C4 C_5 mole fraction	0.06538	0.02080	0.00500[a]	0.00513[a]
LT-ISO Reid vapour pressure (psi)	14.000[b]	12.185	11.563	11.117
LT-ISO bubble point temperature (°F)	116.54	124.56	127.57	129.87
HV-ISO 10% TBP temperature (°F)	195.23	199.55	191.67	196.92
HV-ISO 90% TBP temperature (°F)	335.84	348.39	349.79	348.03
LT-DIST 10% TBP temperature (°F)	312.59	323.86	320.67	314.69
LT-DIST 90% TBP temperature (°F)	481.35	549.80	553.93[b]	553.96[b]

continued on the next page ...

[a]constraint lower bound active
[b]constraint upper bound active

... Table A.8 continued from the previous page

Cycle	1	2	3	4
Product Standard Volume Flows After Economic Optimization:				
Vapour Product (bbl/day)	992.20	976.49	948.96	930.38
C_4-Distillate Product (bbl/day)	876.60	895.74	910.08	927.94
Light Iso Product (bbl/day)	1623.75	1673.87	1749.32	1831.18
Heavy Iso Product (bbl/day)	3711.10	3800.22	3663.05	3410.52
Light Distillate Product (bbl/day)	2442.23	3358.07	3836.45	4233.53
Splitter Bottoms Product (bbl/day)	12290.37	11250.13	10765.46	10484.49
Reboiler Heat Duties After Economic Optimization:				
Stabilizer (C1) Reboiler (MMBTU/hr)	21.7598	24.6837	27.4311	30.1392
Splitter (C2) Reboiler (MMBTU/hr)	18.8338	15.8092	16.2123	16.0287
Economic Optimization Marginal Stream Values:				
Light Iso Marginal Stream Value ($\$/m^3$)	182.78	185.03	185.80	186.35
Data Reconciliation:				
Final objective function value	47.154	11.932	11.915	11.759
Initial objective function value	257.281	31.391	31.394	31.395
Data reconciliation iterations	16	20	28	19

Table A.9: Winter with no measurement noise ASPEN PLUS v9.3 real time optimizer run.

Cycle	1	2	3	4
Economic Optimization:				
Profit ($/hr)	-170.90	118.37	257.33	356.25
Optimization iterations	4	5	6	5
Total product values ($/hr)	22958.75	23216.48	23356.77	23465.89
Total feed cost ($/hr)	23009.43	22983.67	22983.16	22996.60
Total utility costs ($/hr)	120.21	114.44	116.27	113.04
Economic Optimization Set Points:				
C1 reflux flow (bbl/day)	7012.25	6951.80	6926.19	6845.08
C1 tray 3 temperature (°F)	172.17	169.36	165.32	163.36
DIST-C4 C_5 mole fraction	0.06538	0.05396	0.03911	0.03305
C1 reboiler temperature (°F)	587.59	588.18	583.72	584.78
C2 reflux flow (bbl/day)	14212.28	14084.92	13764.29	13715.39
C4 feed flow (bbl/day)	5647.83	5860.43	5594.30	5373.38
C5 feed flow (bbl/day)	3586.57	4043.16	4240.16	4429.91
C2 reboiler temperature (°F)	587.59	588.18	583.72	584.78
HV-ISO 10% TBP temperature (°F)	195.23	199.97	192.49	195.83
HV-ISO 90% TBP temperature (°F)	335.84	349.58	349.12	344.96
LT-DIST 10% TBP temperature (°F)	312.59	326.12	320.73	314.48
LT-DIST 90% TBP temperature (°F)	481.35	547.24	552.99	552.69
Economic Optimization Inequality Constraints:				
C1 reflux flow (bbl/day)	7012.25	6951.80	6926.19	6845.08
C2 reflux flow (bbl/day)	14212.28	14084.92	13764.29	13715.39
C4 feed flow (bbl/day)	5647.83	5860.43	5594.30	5373.38
C5 feed flow (bbl/day)	3586.57	4043.16	4240.16	4429.91
C1 tray 3 temperature (°F)	172.17	169.36	165.32	163.36
C1 reboiler temperature (°F)	587.59	588.18	583.72	584.78
C2 tray 2 temperature (°F)	151.82	153.16	154.97	156.74
C2 reboiler temperature (°F)	652.65	657.52	662.68	659.93
DIST-C4 C_5 mole fraction	0.06538	0.05396	0.03911	0.03305
LT-ISO Reid vapour pressure (psi)	14.000[a]	13.994[a]	14.000[a]	13.964[a]
LT-ISO bubble point temperature (°F)	116.54	116.59	116.59	116.77
HV-ISO 10% TBP temperature (°F)	195.23	199.97	192.49	195.83
HV-ISO 90% TBP temperature (°F)	335.84	349.58	349.12	344.96
LT-DIST 10% TBP temperature (°F)	312.59	326.12	320.73	314.48
LT-DIST 90% TBP temperature (°F)	481.35	547.24	552.99	552.69

continued on the next page ...

[a]constraint upper bound active

... Table A.9 continued from the previous page

Cycle	1	2	3	4
Product Standard Volume Flows After Economic Optimization:				
Vapour Product (bbl/day)	992.20	931.06	863.01	811.62
C_4-Distillate Product (bbl/day)	876.60	914.42	951.31	986.46
Light Iso Product (bbl/day)	1623.75	1699.51	1815.79	1905.06
Heavy Iso Product (bbl/day)	3711.10	3862.46	3681.87	3442.38
Light Distillate Product (bbl/day)	2442.23	3180.03	3478.59	3713.53
Splitter Bottoms Product (bbl/day)	12290.37	11324.16	11120.72	11064.50
Reboiler Heat Duties After Economic Optimization:				
Stabilizer (C1) Reboiler (MMBTU/hr)	21.7598	21.5004	21.0765	20.7998
Splitter (C2) Reboiler (MMBTU/hr)	18.8338	17.1447	18.1885	17.3737
Economic Optimization Marginal Stream Values:				
Light Iso Marginal Stream Value ($/m^3)	182.78	182.78	182.77	182.82

continued on the next page ...

... Table A.9 continued from the previous page

Cycle	5	6	7	8
Economic Optimization:				
Profit ($/hr)	449.50	538.78	646.13	720.02
Optimization iterations	7	13	5	15
Total product values ($/hr)	23573.27	23662.99	23754.80	23809.58
Total feed cost ($/hr)	23009.63	23007.08	22995.96	22974.35
Total utility costs ($/hr)	114.14	117.13	112.70	115.22
Economic Optimization Set Points:				
C1 reflux flow (bbl/day)	6759.18	6645.49	6638.26	6537.43
C1 tray 3 temperature (°F)	160.31	161.51	154.13	153.76
DIST-C4 C_5 mole fraction	0.02418	0.02849	0.00885	0.00874
C1 reboiler temperature (°F)	584.02	581.32	576.67	572.41
C2 reflux flow (bbl/day)	13544.79	13276.08	13022.14	12685.23
C4 feed flow (bbl/day)	5330.80	5309.43	5035.32	5026.37
C5 feed flow (bbl/day)	4642.20	4887.17	5090.37	5223.63
C2 reboiler temperature (°F)	584.02	581.32	576.67	572.41
HV-ISO 10% TBP temperature (°F)	198.37	203.23	203.35	210.34
HV-ISO 90% TBP temperature (°F)	342.26	343.01	334.07	335.97
LT-DIST 10% TBP temperature (°F)	311.02	311.53	302.00	302.03
LT-DIST 90% TBP temperature (°F)	553.83	554.11	554.00	554.16
Economic Optimization Inequality Constraints:				
C1 reflux flow (bbl/day)	6759.18	6645.49	6638.26	6537.43
C2 reflux flow (bbl/day)	13544.79	13276.08	13022.14	12685.23
C4 feed flow (bbl/day)	5330.80	5309.43	5035.32	5026.37
C5 feed flow (bbl/day)	4642.20	4887.17	5090.37	5223.63
C1 tray 3 temperature (°F)	160.31	161.51	154.13	153.76
C1 reboiler temperature (°F)	584.02	581.32	576.67	572.41
C2 tray 2 temperature (°F)	159.81	165.57	166.69	171.45
C2 reboiler temperature (°F)	664.86	670.51	667.50	671.91
DIST-C4 C_5 mole fraction	0.02418	0.02849	0.00885	0.00874
LT-ISO Reid vapour pressure (psi)	14.054[a]	13.529[a]	14.001[a]	13.608
LT-ISO bubble point temperature (°F)	116.41	118.77	116.70	118.49
HV-ISO 10% TBP temperature (°F)	198.37	203.23	203.35	210.34
HV-ISO 90% TBP temperature (°F)	342.26	343.01	334.07	335.97
LT-DIST 10% TBP temperature (°F)	311.02	311.53	302.00[b]	302.03[b]
LT-DIST 90% TBP temperature (°F)	553.83[a]	554.11[a]	554.00[a]	554.16[a]

continued on the next page ...

[a]constraint upper bound active
[b]constraint lower bound active

... Table A.9 continued from the previous page

Cycle	5	6	7	8
Product Standard Volume Flows After Economic Optimization:				
Vapour Product (bbl/day)	759.06	713.78	654.86	612.84
C_4-Distillate Product (bbl/day)	1013.00	1063.51	1068.88	1102.80
Light Iso Product (bbl/day)	2007.02	2112.60	2215.71	2343.01
Heavy Iso Product (bbl/day)	3269.41	3138.29	2791.11	2657.43
Light Distillate Product (bbl/day)	3896.56	4083.66	4335.68	4447.03
Splitter Bottoms Product (bbl/day)	10992.02	10822.20	10857.30	10739.76
Reboiler Heat Duties After Economic Optimization:				
Stabilizer (C1) Reboiler (MMBTU/hr)	20.3736	20.3296	19.7150	19.5143
Splitter (C2) Reboiler (MMBTU/hr)	18.1690	19.2232	18.3444	19.3937
Economic Optimization Marginal Stream Values:				
Light Iso Marginal Stream Value ($/m^3)	182.71	183.36	182.77	183.26

continued on the next page ...

... Table A.9 continued from the previous page

Cycle	9	10
Economic Optimization:		
Profit ($/hr)	816.64	916.62
Optimization iterations	7	10
Total product values ($/hr)	23920.16	24037.75
Total feed cost ($/hr)	22987.14	23003.05
Total utility costs ($/hr)	116.38	118.07
Economic Optimization Set Points:		
C1 reflux flow (bbl/day)	6471.86	6292.07
C1 tray 3 temperature (°F)	151.90	154.94
DIST-C4 C_5 mole fraction	0.00535	0.01307
C1 reboiler temperature (°F)	571.09	576.04
C2 reflux flow (bbl/day)	12436.63	12223.36
C4 feed flow (bbl/day)	4816.08	4617.93
C5 feed flow (bbl/day)	5366.10	5549.39
C2 reboiler temperature (°F)	571.09	576.04
HV-ISO 10% TBP temperature (°F)	218.93	216.49
HV-ISO 90% TBP temperature (°F)	338.82	343.15
LT-DIST 10% TBP temperature (°F)	302.03	302.00
LT-DIST 90% TBP temperature (°F)	554.05	553.99
Economic Optimization Inequality Constraints:		
C1 reflux flow (bbl/day)	6471.86	6292.07
C2 reflux flow (bbl/day)	12436.63	12223.36
C4 feed flow (bbl/day)	4816.08	4617.93
C5 feed flow (bbl/day)	5366.10	5549.39
C1 tray 3 temperature (°F)	151.90	154.94
C1 reboiler temperature (°F)	571.09	576.04
C2 tray 2 temperature (°F)	174.67	179.38
C2 reboiler temperature (°F)	676.82	682.26
DIST-C4 C_5 mole fraction	0.00535[a]	0.01307
LT-ISO Reid vapour pressure (psi)	13.535	12.653
LT-ISO bubble point temperature (°F)	118.85	123.06
HV-ISO 10% TBP temperature (°F)	218.93	216.49
HV-ISO 90% TBP temperature (°F)	338.82	343.15
LT-DIST 10% TBP temperature (°F)	302.03[a]	302.00[a]
LT-DIST 90% TBP temperature (°F)	554.05[b]	553.99[b]

continued on the next page ...

[a]constraint lower bound active
[b]constraint upper bound active

... Table A.9 continued from the previous page

Cycle	9	10
Product Standard Volume Flows After Economic Optimization:		
Vapour Product (bbl/day)	573.19	544.10
C_4-Distillate Product (bbl/day)	1123.05	1176.63
Light Iso Product (bbl/day)	2470.17	2601.60
Heavy Iso Product (bbl/day)	2547.30	2385.88
Light Distillate Product (bbl/day)	4636.35	4879.34
Splitter Bottoms Product (bbl/day)	10565.05	10342.68
Reboiler Heat Duties After Economic Optimization:		
Stabilizer (C1) Reboiler (MMBTU/hr)	19.1953	19.4916
Splitter (C2) Reboiler (MMBTU/hr)	20.1041	20.3809
Economic Optimization Marginal Stream Values:		
Light Iso Marginal Stream Value ($\$/m^3$)	183.35	184.45

Table A.10: Winter to summer ASPEN PLUS v9.3 real time optimizer run.

	Winter Prices		Summer Prices
Cycle	**1**	**2**	**3**
Economic Optimization:			
Profit ($/hr)	-170.90	137.13	196.95
Optimization iterations	4	5	10
Total product values ($/hr)	22958.75	23285.14	23270.31
Total feed cost ($/hr)	23009.43	23028.10	22941.86
Total utility costs ($/hr)	120.21	119.91	131.50
Economic Optimization Set Points:			
C1 reflux flow (bbl/day)	7012.25	7569.65	7907.23
C1 tray 3 temperature (°F)	172.17	161.52	176.08
DIST-C4 C_5 mole fraction	0.06538	0.02080	0.07663
C1 reboiler temperature (°F)	587.59	597.02	606.39
C2 reflux flow (bbl/day)	14212.28	14207.00	14076.68
C4 feed flow (bbl/day)	5647.83	5951.76	5841.89
C5 feed flow (bbl/day)	3586.57	4149.26	4395.64
C2 reboiler temperature (°F)	587.59	597.02	606.39
HV-ISO 10% TBP temperature (°F)	195.23	199.55	199.79
HV-ISO 90% TBP temperature (°F)	335.84	348.39	352.86
LT-DIST 10% TBP temperature (°F)	312.59	323.86	324.26
LT-DIST 90% TBP temperature (°F)	481.35	549.80	554.01
Economic Optimization Inequality Constraints:			
C1 reflux flow (bbl/day)	7012.25	7569.65	7907.23
C2 reflux flow (bbl/day)	14212.28	14207.00	14076.68
C4 feed flow (bbl/day)	5647.83	5951.76	5841.89
C5 feed flow (bbl/day)	3586.57	4149.26	4395.64
C1 tray 3 temperature (°F)	172.17	161.52	176.08
C1 reboiler temperature (°F)	587.59	597.02	606.39
C2 tray 2 temperature (°F)	151.82	154.51	162.92
C2 reboiler temperature (°F)	652.65	651.52	660.82
DIST-C4 C_5 mole fraction	0.06538	0.02080	0.07663
LT-ISO Reid vapour pressure (psi)	14.000[a]	12.185	10.500[a]
LT-ISO bubble point temperature (°F)	116.54	124.56	133.24
HV-ISO 10% TBP temperature (°F)	195.23	199.55	199.79
HV-ISO 90% TBP temperature (°F)	335.84	348.39	352.86
LT-DIST 10% TBP temperature (°F)	312.59	323.86	324.26
LT-DIST 90% TBP temperature (°F)	481.35	549.80	554.01[a]

continued on the next page ...

[a]constraint upper bound active

... Table A.10 continued from the previous page

	Winter Prices		Summer Prices
Cycle	**1**	**2**	**3**
Product Standard Volume Flows After Economic Optimization:			
Vapour Product (bbl/day)	992.20	976.49	997.81
C_4-Distillate Product (bbl/day)	876.60	895.74	962.61
Light Iso Product (bbl/day)	1623.75	1673.87	1763.51
Heavy Iso Product (bbl/day)	3711.10	3800.22	3653.14
Light Distillate Product (bbl/day)	2442.23	3358.07	3734.30
Splitter Bottoms Product (bbl/day)	12290.37	11250.13	10760.98
Reboiler Heat Duties After Economic Optimization:			
Stabilizer (C1) Reboiler (MMBTU/hr)	21.7598	24.6837	28.4160
Splitter (C2) Reboiler (MMBTU/hr)	18.8338	15.8092	15.9911
Economic Optimization Marginal Stream Values:			
Light Iso Marginal Stream Value ($/m^3)	182.78	185.03	187.12
Data Reconciliation:			
Final objective function value	47.154	11.932	11.915
Initial objective function value	257.281	31.391	31.394
Data reconciliation iterations	16	20	28

continued on the next page ...

... Table A.10 continued from the previous page

	Summer Prices	
Cycle	**4**	**5**
Economic Optimization:		
Profit ($/hr)	179.70	228.91
Optimization iterations	4	20
Total product values ($/hr)	23273.10	23306.61
Total feed cost ($/hr)	22954.41	22927.49
Total utility costs ($/hr)	138.99	150.22
Economic Optimization Set Points:		
C1 reflux flow (bbl/day)	8337.80	8958.78
C1 tray 3 temperature (°F)	181.82	179.79
DIST-C4 C_5 mole fraction	0.10366	0.09002
C1 reboiler temperature (°F)	616.01	620.57
C2 reflux flow (bbl/day)	14409.65	14566.67
C4 feed flow (bbl/day)	5982.55	6326.35
C5 feed flow (bbl/day)	4304.87	4296.98
C2 reboiler temperature (°F)	616.01	620.57
HV-ISO 10% TBP temperature (°F)	196.74	198.91
HV-ISO 90% TBP temperature (°F)	354.82	358.23
LT-DIST 10% TBP temperature (°F)	328.63	331.24
LT-DIST 90% TBP temperature (°F)	556.56	553.82
Economic Optimization Inequality Constraints:		
C1 reflux flow (bbl/day)	8337.80	8958.78
C2 reflux flow (bbl/day)	14409.65	14566.67
C4 feed flow (bbl/day)	5982.55	6326.35
C5 feed flow (bbl/day)	4304.87	4296.98
C1 tray 3 temperature (°F)	181.82	179.79
C1 reboiler temperature (°F)	616.01	620.57
C2 tray 2 temperature (°F)	161.63	161.58
C2 reboiler temperature (°F)	655.83	662.96
DIST-C4 C_5 mole fraction	0.10366	0.09002
LT-ISO Reid vapour pressure (psi)	10.441	10.491[a]
LT-ISO bubble point temperature (°F)	133.53	133.25
HV-ISO 10% TBP temperature (°F)	196.74	198.91
HV-ISO 90% TBP temperature (°F)	354.82	358.23
LT-DIST 10% TBP temperature (°F)	328.63	331.24
LT-DIST 90% TBP temperature (°F)	556.56[a]	553.82[a]

continued on the next page ...

[a]constraint upper bound active

... Table A.10 continued from the previous page

	Summer Prices	
Cycle	**4**	**5**
Product Standard Volume Flows After Economic Optimization:		
Vapour Product (bbl/day)	999.93	970.09
C_4-Distillate Product (bbl/day)	1005.07	1013.80
Light Iso Product (bbl/day)	1658.41	1690.58
Heavy Iso Product (bbl/day)	3822.38	3924.78
Light Distillate Product (bbl/day)	3663.98	3680.88
Splitter Bottoms Product (bbl/day)	10733.18	10577.61
Reboiler Heat Duties After Economic Optimization:		
Stabilizer (C1) Reboiler (MMBTU/hr)	31.9781	34.6356
Splitter (C2) Reboiler (MMBTU/hr)	14.9571	16.0918
Economic Optimization Marginal Stream Values:		
Light Iso Marginal Stream Value ($/m^3)	187.19	187.13
Data Reconciliation:		
Final objective function value	11.849	12.459
Initial objective function value	31.393	31.397
Data reconciliation iterations	18	19

Table A.11: Winter to summer with no measurement noise ASPEN PLUS v9.3 real time optimizer run.

	Winter Prices			Summer Prices
Cycle	**1**	**2**	**3**	**4**
Economic Optimization:				
Profit ($/hr)	-170.90	118.37	257.33	265.14
Optimization iterations	4	5	6	6
Total product values ($/hr)	22958.75	23216.48	23356.77	23381.39
Total feed cost ($/hr)	23009.43	22983.67	22983.16	22996.60
Total utility costs ($/hr)	120.21	114.44	116.27	119.66
Economic Optimization Set Points:				
C1 reflux flow (bbl/day)	7012.25	6951.80	6926.19	6919.29
C1 tray 3 temperature (°F)	172.17	169.36	165.32	173.16
DIST-C4 C_5 mole fraction	0.06538	0.05396	0.03911	0.06974
C1 reboiler temperature (°F)	587.59	588.18	583.72	588.41
C2 reflux flow (bbl/day)	14212.28	14084.92	13764.29	13600.06
C4 feed flow (bbl/day)	5647.83	5860.43	5594.30	5675.63
C5 feed flow (bbl/day)	3586.57	4043.16	4240.16	4133.95
C2 reboiler temperature (°F)	587.59	588.18	583.72	588.41
HV-ISO 10% TBP temperature (°F)	195.23	199.97	192.49	204.85
HV-ISO 90% TBP temperature (°F)	335.84	349.58	349.12	358.03
LT-DIST 10% TBP temperature (°F)	312.59	326.12	320.73	332.32
LT-DIST 90% TBP temperature (°F)	481.35	547.24	552.99	557.70
Economic Optimization Inequality Constraints:				
C1 reflux flow (bbl/day)	7012.25	6951.80	6926.19	6919.29
C2 reflux flow (bbl/day)	14212.28	14084.92	13764.29	13600.06
C4 feed flow (bbl/day)	5647.83	5860.43	5594.30	5675.63
C5 feed flow (bbl/day)	3586.57	4043.16	4240.16	4133.95
C1 tray 3 temperature (°F)	172.17	169.36	165.32	173.16
C1 reboiler temperature (°F)	587.59	588.18	583.72	588.41
C2 tray 2 temperature (°F)	151.82	153.16	154.97	164.89
C2 reboiler temperature (°F)	652.65	657.52	662.68	668.04
DIST-C4 C_5 mole fraction	0.06538	0.05396	0.03911	0.06974
LT-ISO Reid vapour pressure (psi)	14.000[a]	13.994[a]	14.000[a]	12.185[b]
LT-ISO bubble point temperature (°F)	116.54	116.59	116.59	124.93
HV-ISO 10% TBP temperature (°F)	195.23	199.97	192.49	204.85
HV-ISO 90% TBP temperature (°F)	335.84	349.58	349.12	358.03
LT-DIST 10% TBP temperature (°F)	312.59	326.12	320.73	332.32
LT-DIST 90% TBP temperature (°F)	481.35	547.24	552.99	557.70[b]

continued on the next page ...

[a]constraint upper bound active
[b]constraint upper bound violated

... Table A.11 continued from the previous page

	Winter Prices			Summer Prices
Cycle	**1**	**2**	**3**	**4**
Product Standard Volume Flows After Economic Optimization:				
Vapour Product (bbl/day)	992.20	931.06	863.01	951.63
C_4-Distillate Product (bbl/day)	876.60	914.42	951.31	930.78
Light Iso Product (bbl/day)	1623.75	1699.51	1815.79	1930.93
Heavy Iso Product (bbl/day)	3711.10	3862.46	3681.87	3834.66
Light Distillate Product (bbl/day)	2442.23	3180.03	3478.59	3383.02
Splitter Bottoms Product (bbl/day)	12290.37	11324.16	11120.72	10892.90
Reboiler Heat Duties After Economic Optimization:				
Stabilizer (C1) Reboiler (MMBTU/hr)	21.7598	21.5004	21.0765	22.0054
Splitter (C2) Reboiler (MMBTU/hr)	18.8338	17.1447	18.1885	18.4016
Economic Optimization Marginal Stream Values:				
Light Iso Marginal Stream Value ($/m^3)	182.78	182.78	182.77	185.03

continued on the next page ...

... Table A.11 continued from the previous page

	Summer Prices			
Cycle	**5**	**6**	**7**	**8**
Economic Optimization:				
Profit ($/hr)	253.60	335.06	399.57	476.62
Optimization iterations	11	6	9	12
Total product values ($/hr)	23354.58	23441.45	23504.31	23602.23
Total feed cost ($/hr)	22977.60	22983.47	22980.93	22997.50
Total utility costs ($/hr)	123.38	122.92	123.81	128.11
Economic Optimization Set Points:				
C1 reflux flow (bbl/day)	6828.33	6733.51	6629.28	6539.20
C1 tray 3 temperature (°F)	180.41	178.63	177.62	177.22
DIST-C4 C_5 mole fraction	0.10436	0.09520	0.09027	0.08817
C1 reboiler temperature (°F)	593.55	588.67	589.16	585.82
C2 reflux flow (bbl/day)	13268.17	13022.09	12832.62	12662.60
C4 feed flow (bbl/day)	5734.76	5724.41	5531.47	5799.03
C5 feed flow (bbl/day)	4060.18	3913.34	4084.16	3939.03
C2 reboiler temperature (°F)	593.55	588.67	589.16	585.82
HV-ISO 10% TBP temperature (°F)	215.65	221.73	228.51	223.07
HV-ISO 90% TBP temperature (°F)	366.13	374.91	376.10	387.78
LT-DIST 10% TBP temperature (°F)	337.53	342.12	340.75	349.64
LT-DIST 90% TBP temperature (°F)	554.13	554.64	553.86	553.98
Economic Optimization Inequality Constraints:				
C1 reflux flow (bbl/day)	6828.33	6733.51	6629.28	6539.20
C2 reflux flow (bbl/day)	13268.17	13022.09	12832.62	12662.60
C4 feed flow (bbl/day)	5734.76	5724.41	5531.47	5799.03
C5 feed flow (bbl/day)	4060.18	3913.34	4084.16	3939.03
C1 tray 3 temperature (°F)	180.41	178.63	177.62	177.22
C1 reboiler temperature (°F)	593.55	588.67	589.16	585.82
C2 tray 2 temperature (°F)	175.62	178.01	180.87	183.70
C2 reboiler temperature (°F)	673.19	677.92	682.46	688.11
DIST-C4 C_5 mole fraction	0.10436	0.09520	0.09027	0.08817
LT-ISO Reid vapour pressure (psi)	10.500[a]	10.588[b]	10.495[a]	10.331
LT-ISO bubble point temperature (°F)	133.85	133.49	134.15	135.21
HV-ISO 10% TBP temperature (°F)	215.65	221.73	228.51	223.07
HV-ISO 90% TBP temperature (°F)	366.13	374.91	376.10	387.78
LT-DIST 10% TBP temperature (°F)	337.53	342.12	340.75	349.64
LT-DIST 90% TBP temperature (°F)	554.13[a]	554.64[a]	553.86[a]	553.98[a]

continued on the next page ...

[a]constraint upper bound active
[b]constraint upper bound violated

... Table A.11 continued from the previous page

	Summer Prices			
Cycle	**5**	**6**	**7**	**8**
Product Standard Volume Flows After Economic Optimization:				
Vapour Product (bbl/day)	978.44	912.94	855.61	801.29
C_4-Distillate Product (bbl/day)	977.83	1021.41	1065.87	1115.14
Light Iso Product (bbl/day)	2051.35	2155.37	2267.20	2381.49
Heavy Iso Product (bbl/day)	3826.05	3930.12	3797.74	3971.45
Light Distillate Product (bbl/day)	3317.84	3226.62	3433.12	3281.81
Splitter Bottoms Product (bbl/day)	10754.64	10665.10	10489.92	10373.86
Reboiler Heat Duties After Economic Optimization:				
Stabilizer (C1) Reboiler (MMBTU/hr)	22.8092	22.3782	22.1024	21.8891
Splitter (C2) Reboiler (MMBTU/hr)	18.8549	19.1314	19.7082	21.3738
Economic Optimization Marginal Stream Values:				
Light Iso Marginal Stream Value ($/m^3)	187.12	187.01	187.13	187.33

continued on the next page ...

... Table A.11 continued from the previous page

	Summer Prices	
Cycle	**9**	**10**
Economic Optimization:		
Profit ($/hr)	503.70	580.42
Optimization iterations	5	8
Total product values ($/hr)	23622.12	23713.34
Total feed cost ($/hr)	22986.97	23000.39
Total utility costs ($/hr)	131.45	132.54
Economic Optimization Set Points:		
C1 reflux flow (bbl/day)	6356.84	6360.76
C1 tray 3 temperature (°F)	181.75	177.59
DIST-C4 C_5 mole fraction	0.11016	0.08957
C1 reboiler temperature (°F)	588.60	583.88
C2 reflux flow (bbl/day)	12448.03	12144.23
C4 feed flow (bbl/day)	5727.36	5697.30
C5 feed flow (bbl/day)	4150.50	4001.39
C2 reboiler temperature (°F)	588.60	583.88
HV-ISO 10% TBP temperature (°F)	231.23	234.67
HV-ISO 90% TBP temperature (°F)	390.43	394.86
LT-DIST 10% TBP temperature (°F)	354.17	357.47
LT-DIST 90% TBP temperature (°F)	554.64	553.74
Economic Optimization Inequality Constraints:		
C1 reflux flow (bbl/day)	6356.84	6360.76
C2 reflux flow (bbl/day)	12448.03	12144.23
C4 feed flow (bbl/day)	5727.36	5697.30
C5 feed flow (bbl/day)	4150.50	4001.39
C1 tray 3 temperature (°F)	181.75	177.59
C1 reboiler temperature (°F)	588.60	583.88
C2 tray 2 temperature (°F)	190.85	192.95
C2 reboiler temperature (°F)	692.96	695.99
DIST-C4 C_5 mole fraction	0.11016	0.08957
LT-ISO Reid vapour pressure (psi)	9.329	9.768
LT-ISO bubble point temperature (°F)	141.43	138.87
HV-ISO 10% TBP temperature (°F)	231.23	234.67
HV-ISO 90% TBP temperature (°F)	390.43	394.86
LT-DIST 10% TBP temperature (°F)	354.17	357.47
LT-DIST 90% TBP temperature (°F)	554.64[a]	553.74[a]

continued on the next page ...

[a]constraint upper bound active

... Table A.11 continued from the previous page

	Winter Prices	
Cycle	**9**	**10**
Product Standard Volume Flows After Economic Optimization:		
Vapour Product (bbl/day)	779.36	720.09
C_4-Distillate Product (bbl/day)	1190.79	1196.45
Light Iso Product (bbl/day)	2503.26	2632.64
Heavy Iso Product (bbl/day)	3851.53	3919.91
Light Distillate Product (bbl/day)	3427.78	3325.40
Splitter Bottoms Product (bbl/day)	10161.71	10133.13
Reboiler Heat Duties After Economic Optimization:		
Stabilizer (C1) Reboiler (MMBTU/hr)	22.3948	21.6311
Splitter (C2) Reboiler (MMBTU/hr)	21.9934	23.1252
Economic Optimization Marginal Stream Values:		
Light Iso Marginal Stream Value ($/m^3)	188.57	188.03

Table A.12: ASPEN PLUS real time optimizer run with external condendser disturbances.

Cycle	1	2	3	4
Economic Optimization:				
Profit ($/hr)	-9.84	204.71	136.48	103.64
Optimization iterations	10	7	14	6
Total product values ($/hr)	23035.02	23124.08	23027.42	22988.36
Total feed cost ($/hr)	22928.41	22796.18	22762.67	22751.29
Total utility costs ($/hr)	116.45	123.19	128.27	133.43
Economic Optimization Set Points:				
C1 reflux flow (bbl/day)	6913.71	7574.51	8090.61	8673.31
C1 tray 3 temperature (°F)	181.26	173.04	174.54	173.11
DIST-C4 C_5 mole fraction	0.11882	0.06681	0.07033	0.06068
C1 reboiler temperature (°F)	595.10	601.80	612.81	617.63
C2 reflux flow (bbl/day)	13951.69	13967.29	13622.78	13407.55
C4 feed flow (bbl/day)	5998.29	6447.35	6345.58	6072.69
C5 feed flow (bbl/day)	3334.48	3483.00	3236.97	3010.71
C2 reboiler temperature (°F)	595.10	601.80	612.81	617.63
HV-ISO 10% TBP temperature (°F)	197.38	198.46	197.74	193.32
HV-ISO 90% TBP temperature (°F)	351.62	368.63	367.62	365.04
LT-DIST 10% TBP temperature (°F)	338.26	355.20	359.43	357.30
LT-DIST 90% TBP temperature (°F)	496.25	554.11	554.10	552.19
Economic Optimization Inequality Constraints:				
C1 reflux flow (bbl/day)	6913.71	7574.51	8090.61	8673.31
C2 reflux flow (bbl/day)	13951.69	13967.29	13622.78	13407.55
C4 feed flow (bbl/day)	5998.29	6447.35	6345.58	6072.69
C5 feed flow (bbl/day)	3334.48	3483.00	3236.97	3010.71
C1 tray 3 temperature (°F)	181.26	173.04	174.54	173.11
C1 reboiler temperature (°F)	595.10	601.80	612.81	617.63
C2 tray 2 temperature (°F)	153.02	153.83	154.55	151.63
C2 reboiler temperature (°F)	651.22	663.09	645.76	634.25
DIST-C4 C_5 mole fraction	0.11882	0.06681	0.07033	0.06068
LT-ISO Reid vapour pressure (psi)	14.000[a]	12.436	11.659	11.988
LT-ISO bubble point temperature (°F)	115.05	121.22	124.63	122.72
HV-ISO 10% TBP temperature (°F)	197.38	198.46	197.74	193.32
HV-ISO 90% TBP temperature (°F)	351.62	368.63	367.62	365.04
LT-DIST 10% TBP temperature (°F)	338.26	355.20	359.43	357.30
LT-DIST 90% TBP temperature (°F)	496.25	554.11[a]	554.10[a]	552.19

continued on the next page ...

[a]constraint upper bound active

... Table A.12 continued from the previous page

Cycle	1	2	3	4
Product Standard Volume Flows After Economic Optimization:				
Vapour Product (bbl/day)	1024.90	1001.83	1010.12	993.86
C_4-Distillate Product (bbl/day)	902.05	900.70	921.20	926.90
Light Iso Product (bbl/day)	1559.61	1565.83	1517.12	1446.06
Heavy Iso Product (bbl/day)	4202.01	4627.11	4608.50	4587.07
Light Distillate Product (bbl/day)	1952.44	2474.43	2167.61	2031.50
Splitter Bottoms Product (bbl/day)	12218.00	11161.76	11476.82	11704.48
Reboiler Heat Duties After Economic Optimization:				
Stabilizer (C1) Reboiler (MMBTU/hr)	21.2420	23.3510	26.5021	29.2489
Splitter (C2) Reboiler (MMBTU/hr)	18.0814	18.2495	16.8149	15.8111
Economic Optimization Marginal Stream Values:				
Light Iso Marginal Stream Value ($\$/m^3$)	182.78	184.72	185.68	185.27
Data Reconciliation:				
Final objective function value	39.732	17.926	13.634	12.716
Initial objective function value	255.659	41.409	42.860	36.767
Data reconciliation iterations	15	11	20	19

Table A.13: ASPEN PLUS real time optimizer run with external condendser disturbances and no measurement noise.

Cycle	1	2	3
Economic Optimization:			
Profit ($/hr)	-9.84	141.59	116.81
Optimization iterations	10	16	7
Total product values ($/hr)	23035.02	23160.33	23114.22
Total feed cost ($/hr)	22928.41	22905.32	22881.38
Total utility costs ($/hr)	116.45	113.42	116.02
Economic Optimization Set Points:			
C1 reflux flow (bbl/day)	6913.71	6727.41	6834.17
C1 tray 3 temperature (°F)	181.26	184.02	185.89
DIST-C4 C_5 mole fraction	0.11882	0.13807	0.15206
C1 reboiler temperature (°F)	595.10	596.31	596.90
C2 reflux flow (bbl/day)	13951.69	13807.65	13747.68
C4 feed flow (bbl/day)	5998.29	6337.05	6495.17
C5 feed flow (bbl/day)	3334.48	3623.67	3457.30
C2 reboiler temperature (°F)	595.10	596.31	596.90
HV-ISO 10% TBP temperature (°F)	197.38	193.81	200.13
HV-ISO 90% TBP temperature (°F)	351.62	361.98	372.89
LT-DIST 10% TBP temperature (°F)	338.26	353.10	366.03
LT-DIST 90% TBP temperature (°F)	496.25	553.68	556.51
Economic Optimization Inequality Constraints:			
C1 reflux flow (bbl/day)	6913.71	6727.41	6834.17
C2 reflux flow (bbl/day)	13951.69	13807.65	13747.68
C4 feed flow (bbl/day)	5998.29	6337.05	6495.17
C5 feed flow (bbl/day)	3334.48	3623.67	3457.30
C1 tray 3 temperature (°F)	181.26	184.02	185.89
C1 reboiler temperature (°F)	595.10	596.31	596.90
C2 tray 2 temperature (°F)	153.02	155.58	157.21
C2 reboiler temperature (°F)	651.22	657.44	661.89
DIST-C4 C_5 mole fraction	0.11882	0.13807	0.15206[a]
LT-ISO Reid vapour pressure (psi)	14.000[a]	13.990[a]	14.028[a]
LT-ISO bubble point temperature (°F)	115.05	115.42	115.42
HV-ISO 10% TBP temperature (°F)	197.38	193.81	200.13
HV-ISO 90% TBP temperature (°F)	351.62	361.98	372.89
LT-DIST 10% TBP temperature (°F)	338.26	353.10	366.03
LT-DIST 90% TBP temperature (°F)	496.25	553.68[a]	556.51[a]

continued on the next page ...

[a]constraint upper bound active

... Table A.13 continued from the previous page

Cycle	1	2	3
Product Standard Volume Flows After Economic Optimization:			
Vapour Product (bbl/day)	1024.90	984.36	1057.79
C_4-Distillate Product (bbl/day)	902.05	957.58	879.47
Light Iso Product (bbl/day)	1559.61	1615.82	1677.26
Heavy Iso Product (bbl/day)	4202.01	4420.61	4653.28
Light Distillate Product (bbl/day)	1952.44	2404.88	2136.40
Splitter Bottoms Product (bbl/day)	12218.00	11453.92	11410.01
Reboiler Heat Duties After Economic Optimization:			
Stabilizer (C1) Reboiler (MMBTU/hr)	21.2420	20.5632	19.8622
Splitter (C2) Reboiler (MMBTU/hr)	18.0814	17.7369	19.3180
Economic Optimization Marginal Stream Values:			
Light Iso Marginal Stream Value ($/m^3)	182.78	182.79	182.74

continued on the next page ...

... Table A.13 continued from the previous page

Cycle	4	5
Economic Optimization:		
Profit ($/hr)	162.95	175.54
Optimization iterations	8	26
Total product values ($/hr)	23179.09	23178.66
Total feed cost ($/hr)	22897.08	22881.78
Total utility costs ($/hr)	119.07	121.34
Economic Optimization Set Points:		
C1 reflux flow (bbl/day)	6740.53	6841.06
C1 tray 3 temperature (°F)	185.62	186.50
DIST-C4 C_5 mole fraction	0.14996	0.15707
C1 reboiler temperature (°F)	594.04	593.93
C2 reflux flow (bbl/day)	13681.67	13543.65
C4 feed flow (bbl/day)	6809.83	7009.66
C5 feed flow (bbl/day)	3304.65	3190.03
C2 reboiler temperature (°F)	594.04	593.93
HV-ISO 10% TBP temperature (°F)	206.14	211.14
HV-ISO 90% TBP temperature (°F)	383.63	394.26
LT-DIST 10% TBP temperature (°F)	375.07	391.09
LT-DIST 90% TBP temperature (°F)	551.84	569.22
Economic Optimization Inequality Constraints:		
C1 reflux flow (bbl/day)	6740.53	6841.06
C2 reflux flow (bbl/day)	13681.67	13543.65
C4 feed flow (bbl/day)	6809.83	7009.66
C5 feed flow (bbl/day)	3304.65	3190.03
C1 tray 3 temperature (°F)	185.62	186.50
C1 reboiler temperature (°F)	594.04	593.93
C2 tray 2 temperature (°F)	158.79	161.65
C2 reboiler temperature (°F)	664.63	668.60
DIST-C4 C_5 mole fraction	0.14996[a]	0.15707[a]
LT-ISO Reid vapour pressure (psi)	14.002[a]	14.070[a]
LT-ISO bubble point temperature (°F)	115.64	115.44
HV-ISO 10% TBP temperature (°F)	206.14	211.14
HV-ISO 90% TBP temperature (°F)	383.63	394.26
LT-DIST 10% TBP temperature (°F)	375.07	391.09
LT-DIST 90% TBP temperature (°F)	551.84	569.22[a]

continued on the next page ...

[a]constraint upper bound active

... Table A.13 continued from the previous page

Cycle	4	5
Product Standard Volume Flows After Economic Optimization:		
Vapour Product (bbl/day)	1009.87	1073.29
C_4-Distillate Product (bbl/day)	921.15	842.52
Light Iso Product (bbl/day)	1745.48	1819.39
Heavy Iso Product (bbl/day)	4857.12	5145.37
Light Distillate Product (bbl/day)	1832.52	1646.03
Splitter Bottoms Product (bbl/day)	11463.31	11287.83
Reboiler Heat Duties After Economic Optimization:		
Stabilizer (C1) Reboiler (MMBTU/hr)	19.5769	18.9294
Splitter (C2) Reboiler (MMBTU/hr)	20.6308	22.0462
Economic Optimization Marginal Stream Values:		
Light Iso Marginal Stream Value ($/m^3)	182.77	182.69

Table A.14: Base Case ASPEN PLUS v9.3 real time optimizer run.

Cycle	1	2	3
Economic Optimization:			
Profit ($/hr)	67.76	241.23	290.75
Optimization iterations	10	6	12
Total product values ($/hr)	23195.72	23329.65	23548.59
Total feed cost ($/hr)	23009.43	22970.32	23130.55
Total utility costs ($/hr)	118.52	118.10	127.30
Economic Optimization Set Points:			
C1 reflux flow (bbl/day)	7012.50	7617.61	8240.18
C1 tray 3 temperature (°F)	172.15	161.30	157.52
DIST-C4 C_5 mole fraction	0.06529	0.01995	0.00589
C1 reboiler temperature (°F)	587.70	594.12	601.79
C2 reflux flow (bbl/day)	14209.57	14331.62	14627.33
C4 feed flow (bbl/day)	6206.91	6617.66	6837.97
C5 feed flow (bbl/day)	3352.91	3476.24	3388.27
C2 reboiler temperature (°F)	587.70	594.12	601.79
HV-ISO 10% TBP temperature (°F)	196.98	192.59	190.67
HV-ISO 90% TBP temperature (°F)	351.81	363.53	370.16
LT-DIST 10% TBP temperature (°F)	336.20	350.82	356.32
LT-DIST 90% TBP temperature (°F)	500.44	553.99	553.97
Economic Optimization Inequality Constraints:			
C1 reflux flow (bbl/day)	7012.50	7617.61	8240.18
C2 reflux flow (bbl/day)	14209.57	14331.62	14627.33
C4 feed flow (bbl/day)	6206.91	6617.66	6837.97
C5 feed flow (bbl/day)	3352.91	3476.24	3388.27
C1 tray 3 temperature (°F)	172.15	161.30	157.52
C1 reboiler temperature (°F)	587.70	594.12	601.79
C2 tray 2 temperature (°F)	151.84	149.13	147.14
C2 reboiler temperature (°F)	652.65	651.44	653.76
DIST-C4 C_5 mole fraction	0.06529	0.01995	0.00589[a]
LT-ISO Reid vapour pressure (psi)	14.000[b]	12.977	12.869
LT-ISO bubble point temperature (°F)	116.54	120.83	121.25
HV-ISO 10% TBP temperature (°F)	196.98	192.59	190.67
HV-ISO 90% TBP temperature (°F)	351.81	363.53	370.16
LT-DIST 10% TBP temperature (°F)	336.20	350.82	356.32
LT-DIST 90% TBP temperature (°F)	500.44	553.99[b]	553.97[b]

continued on the next page ...

[a]constraint lower bound active
[b]constraint upper bound active

... Table A.14 continued from the previous page

Cycle	1	2	3
Product Standard Volume Flows After Economic Optimization:			
Vapour Product (bbl/day)	992.10	969.83	954.94
C_4-Distillate Product (bbl/day)	876.51	894.98	919.96
Light Iso Product (bbl/day)	1623.72	1521.54	1481.84
Heavy Iso Product (bbl/day)	4201.09	4599.76	4864.68
Light Distillate Product (bbl/day)	2124.86	2576.41	2548.58
Splitter Bottoms Product (bbl/day)	12117.93	11336.73	11281.92
Reboiler Heat Duties After Economic Optimization:			
Stabilizer (C1) Reboiler (MMBTU/hr)	21.7576	24.4127	27.2240
Splitter (C2) Reboiler (MMBTU/hr)	18.2673	15.4697	15.7629
Economic Optimization Marginal Stream Values:			
Light Iso Marginal Stream Value ($/m^3)	182.78	184.05	184.18
Data Reconciliation:			
Final objective function value	47.154	11.281	13.312
Initial objective function value	257.281	31.389	31.392
Data reconciliation iterations	16	19	12

NOTE TO USERS

Page(s) missing in number only; text follows. Microfilmed as received.

236

UMI

Appendix B

ASPEN PLUS Real Time Optimizer Documentation

B.1 Purpose

This program performs a given number of cycles of a real time optimizer on the stablizer-splitter flowsheet given in Figure B.1 using ASPEN PLUS to perform the data reconciliation and optimization.

B.2 Specification

To setup the real time optimizer for a run, use the command:

```
setup
```

or

```
nohup setup &
```

This command must be given before running the real time optimizer, and the files listed in Table B.1 must already exist on the paths specified.

To run the real time optimizer for a given number of cycles, n, use the command:

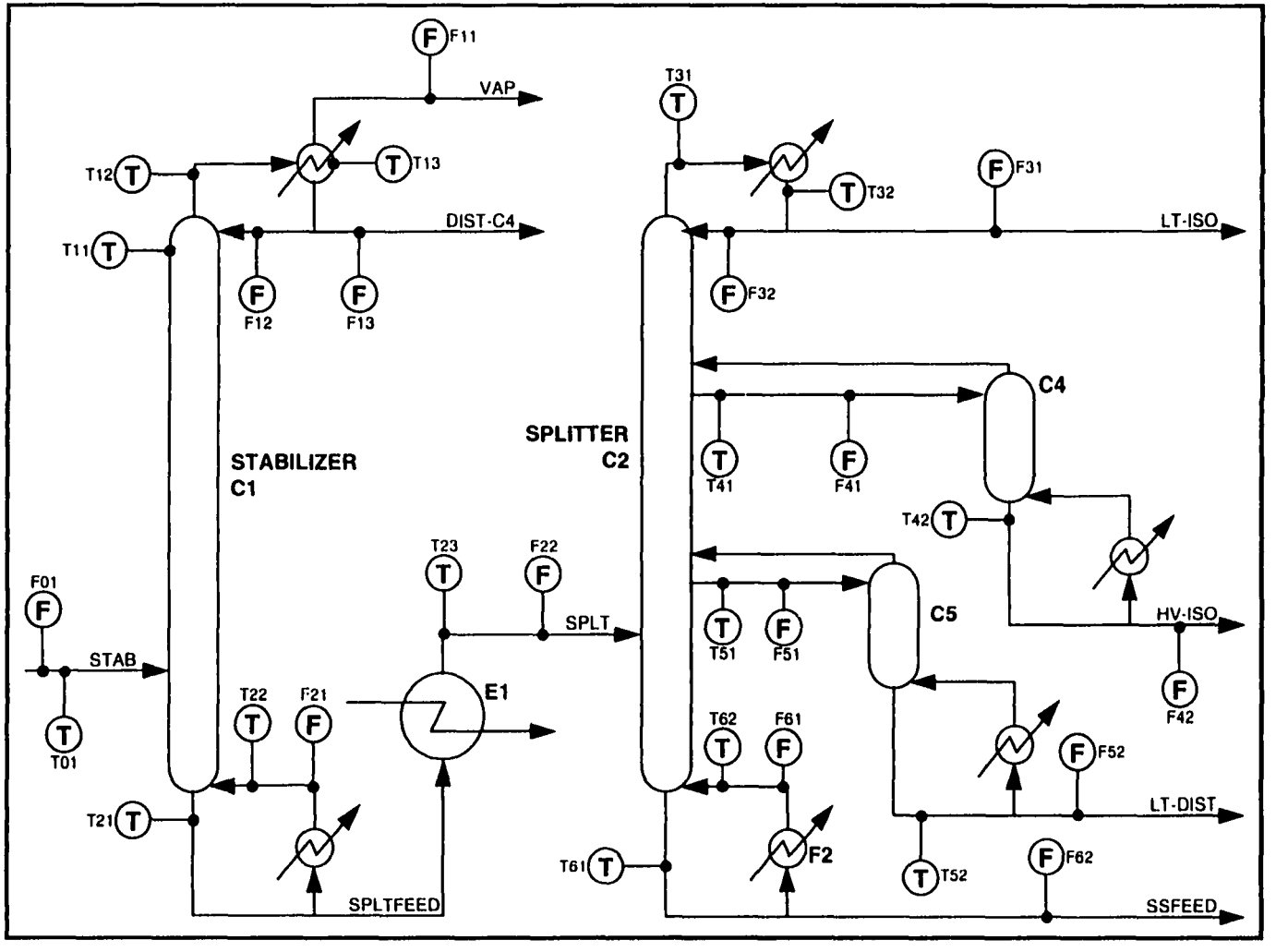


Figure B.1: Stabilizer-splitter process flowsheet (including measurement tag numbers).

`run` n

or

`nohup run` n `&`

The real time optimizer must already have been either `setup` or `run` prior to using this command.

B.3 Description

The ASPEN PLUS real time optimizer performs data reconciliation and optimization on the stabilizer-splitter flowsheet given in Figure B.1. Detailed information about this optimizer is covered by Naysmith [1].

Before running the real time optimizer, the program `setup` must be run first. This program translates and compiles the ASPEN PLUS simulations, and compiles the FORTRAN77 routines used. The files listed in Table B.1 must exist on the paths specified, and the NAG FORTRAN Library Routines must be available. The program `setup` only needs to be run once.

The program `run` runs a specified number of cycles (or iterations) of the real time optimizer. The steps used by this program are summarised in Figure B.2.

The data reconciliation uses a weighted least squares objective function to reconcile 28 measured variables (all variables shown in Figure B.1 except T11):

$$\begin{aligned} &\min_{\mathbf{x}} \left[\sum_{i=1}^{28} w_i \left(\frac{y_i - x_i}{s_i} \right)^2 \right] \\ &\text{subject to } h(\mathbf{x}) = \mathbf{0} \end{aligned} \tag{B.1}$$

where w = weighting factor associated with each measured variable;
y = the measured variable;
x = the estimated true (reconciled) variable value;
s = the standard deviation of the measurements;

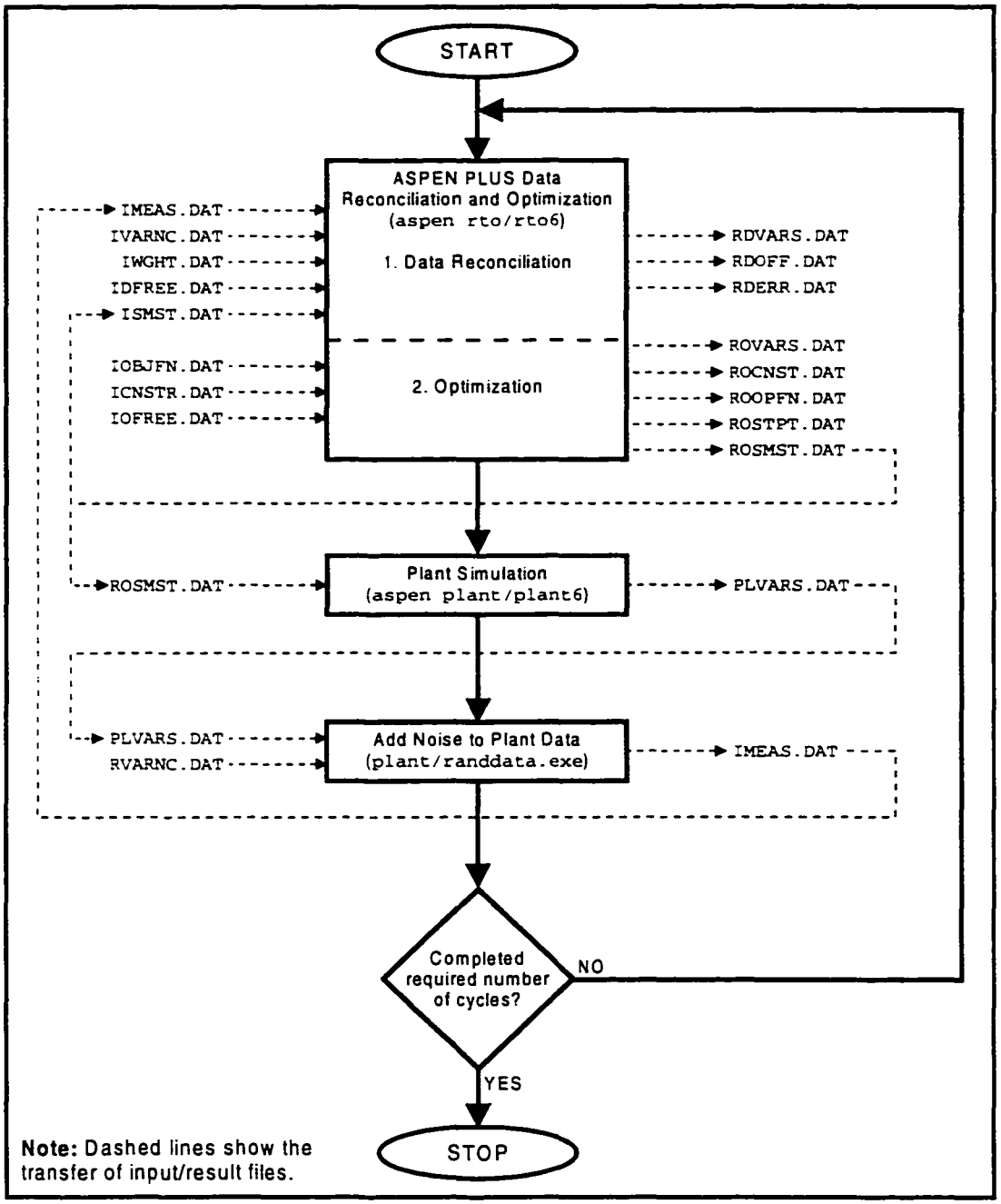


Figure B.2: ASPEN PLUS real time optimization program flow chart.

$h(\mathbf{x})$ = the set of equality constraints representing the process model.

The optimization algorithm used is the built in ASPEN PLUS SQP algorithm. This objective function is minimized by varying 15 free variables.

The economic optimization maximizes profit using the following objective function:

$$\begin{aligned} &\max \left[\sum(\text{product values}) - (\text{feed cost}) - (\text{reboiler fuel oil cost})\right] \\ &\text{subject to } h(\mathbf{x}) = \mathbf{0} \\ &\qquad\qquad g(\mathbf{x}) \geq \mathbf{0} \end{aligned} \tag{B.2}$$

where $h(\mathbf{x})$ = the set of equality constraints representing the process model:
$g(\mathbf{x})$ = product quality and operating inequality constraints.

The optimization algorithm used is the built in ASPEN PLUS SQP algorithm. The objective function is maximized by varying 13 free variables subject to 25 inequality constraints and 2 equality constraints on condenser heat duties.

Normally distributed random noise is added to the plant data calculated in the plant simulation before being returned to the real time optimizer for the next iteration. The noise has a mean of zero, and a given standard deviation for each of the 28 measured variables.

B.4 References

[1] M. Naysmith. Real time optimization of chemical processes — a comparison of closed versus open form equations using commercial simulation packages. Ph. D. Thesis, University of Waterloo, 1997.

B.5 Files

The files required to run ASPEN PLUS real time optimizer are listed in Table B.1.

Table B.1: ASEPN PLUS real time optimizer files required.

File	Description
`run`	Runs the real time optimizer (C-shell script, executable)
`setup`	Sets up the real time optimizer prior to being run (C-shell script, executable)
`plant/plant6.inp`	"Plant" simulation (ASPEN PLUS input file)
`plant/randata.f`	Adds normally distributed random noise to the measured plant data (FORTRAN77 input file, requires the NAG FORTRAN Library Routines)
`plant/RVARNC.DAT`	Contains the random noise standard deviations for the measured plant data
`rto/ICNSTR.DAT`	Contains the optimization inequality constraint upper and lower bounds
`rto/IDFREE.DAT`	Contains the data reconciliation free variable bound ranges
`rto/IMEAS.DAT`	Contains the measured variable data from the plant (with noise added)
`rto/IOBJFN.DAT`	Contains the optimization objective function parameters
`rto/IOFREE.DAT`	Contains the optimization free variable bound ranges
`rto/ISMST.DAT`	Contains the initial simulation settings and parameters for starting the ASPEN PLUS real time optimization simulation
`rto/IVARNC.DAT`	Contains the measured variable standard deviations for the data reconciliation
`rto/IWGHT.DAT`	Contains the measured variable weights for the data reconciliation
`rto/rto6.inp`	Data reconciliation and optimization simulation (ASPEN PLUS input file)

B.6 Input File Specifications

All input files are formatted in the same way using the following FORTRAN format statement:

```
FORMAT(A13, 2X, E15.8)
```

The first 13 columns of each line contain a descriptive label. The next 2 columns are blank and the last 15 columns contain the corresponding numerical value. Example input files are given in section B.8.

B.6.1 IMEAS.DAT, IVARNC.DAT and IWGHT.DAT

`IMEAS.DAT`, `IVARNC.DAT` and `IWGHT.DAT` contain the measured values, standard deviations and weights for data reconciliation corresponding to the 28 measured plant variables respectively. The 28 variables are given in the same order as presented in Table B.2 for all three files. The units given are for the measured variables and standard deviations. The variable names are the same as those used in Figure B.1.

B.6.2 IDFREE.DAT

`IDFREE.DAT` contains the free variable ranges for data reconciliation. The upper and lower are free variable bounds are calculated using the following formulae:

$$\begin{aligned} x_{UB} &= x^{(0)} + r \\ x_{LB} &= x^{(0)} - r \end{aligned} \tag{B.3}$$

where x_{UB} = free variable upper bound;
x_{UB} = free variable lower bound;
$x^{(0)}$ = initial free variable value;
r = free variable range as specified in `IDFREE.DAT`.

The free variable range specifications are given in the order shown in Table B.3.

Table B.2: Variable order used in the ASPEN PLUS real time optimizer input files `IMEAS.DAT`, `IVARNC.DAT` and `IWGHT.DAT`.

Variable	Description	Units
F01	STAB stream standard liquid volume flow	bbl/day
F11	VAP stream standard vapour volume flow	cuft/hr
F12	C1 reflux standard liquid volume flow	bbl/day
F13	DIST-C4 stream standard liquid volume flow	bbl/day
F21	C1 reboiler standard liquid volume flow	bbl/day
F22	E1 outlet standard liquid volume flow	bbl/day
F31	LT-ISO stream standard liquid volume flow	bbl/day
F32	C2 reflux standard liquid volume flow	bbl/day
F41	C4 feed standard liquid volume flow	bbl/day
F42	HV-ISO stream standard liquid volume flow	bbl/day
F51	C5 feed standard liquid volume flow	bbl/day
F52	LT-DIST stream standard liquid volume flow	bbl/day
F61	C2 reboiler standard liquid volume flow	bbl/day
F62	SSFEED stream standard liquid volume flow	bbl/day
T01	STAB stream temperature	°F
T12	C1 tray 2 temperature	°F
T13	C1 condenser temperature	°F
T21	C1 bottoms stream temperature	°F
T22	C1 reboiler outlet temperature	°F
T23	E1 outlet temperature	°F
T31	C2 tray 2 temperature	°F
T32	LT-ISO stream temperature	°F
T41	C4 feed stream temperature	°F
T42	HV-ISO stream temperature	°F
T51	C5 feed stream temperature	°F
T52	LT-DIST stream temperature	°F
T61	C2 bottoms stream temperature	°F
T62	C2 reboiler outlet temperature	°F

Table B.3: Data reconciliation free variable ranges used in the ASPEN PLUS real time optimizer input file IDFREE.DAT.

Variable	Description	Units
STAB-MOLEFLOW	STAB stream mole flow	lbmol/hr
STAB-TEMP	STAB stream temperature	°F
C1-RR	C1 molar reflux ratio	lbmol/lbmol
C1-RDV	C1 condenser molar vapour fraction	lbmol/lbmol
C1-REBL-FLOW	C1 reboiler mole flow	lbmol/hr
C1-REBL-TEMP	C1 reboiler outlet temperature	°F
E1-TEMP	E1 outlet temperature	°F
C2-MOLE-D	C2 liquid distillate mole flow	lbmol/hr
C2-REBL-FLOW	C2 reboiler mole flow	lbmol/hr
C2-REBL-TEMP	C2 reboiler outlet temperature	°F
C2-SC-TEMP	C2 condenser sub-cooled temperature	°F
C4-QN	C4 reboiler heat duty	MMBTU/hr
C5-QN	C5 reboiler heat duty	MMBTU/hr
C2-C4-FLOW	C4 feed stream mole flow	lbmol/hr
C2-C5-FLOW	C5 feed stream mole flow	lbmol/hr

B.6.3 ISMST.DAT

The file `ISMST.DAT` contains all the simulation settings to initialize the ASPEN PLUS real time optimization simulation. Table B.4 summarises the variable in this file. Variables which are vectors, such as compositions and temperature profiles, use n lines in the input file as specified in Table B.4.

B.6.4 IOBJFN.DAT

`IOBJFN.DAT` contains the parameters used in calculating the economic objective function. Table B.5 summarises the order in which these parameters must appear in the file `IOBJFN.DAT`.

B.6.5 ICNSTR.DAT

`ICNSTR.DAT` contains the upper and lower bounds for all the inequality constraints in the optimization. Table B.6 summarises the order in which these bounds must appear in the file `ICNSTR.DAT`.

B.6.6 IOFREE.DAT

`IOFREE.DAT` contains the free variable ranges for optimization. The upper and lower are free variable bounds are calculated using the following formulae for all free variables except temperatures:

$$\begin{aligned} x_{UB} &= x^{(0)} + qx^{(0)} \\ x_{LB} &= x^{(0)} - qx^{(0)} \end{aligned} \tag{B.4}$$

where $q =$ free variable range as a fraction of the initial variable value.

Temperature free variable bounds are calculated using the formulae given in equation (B.3). The free variable range specifications are given in the order shown in Table B.7. Free variable bounds calculated using equation (B.4) have the character '%' at the end of their variable name.

Table B.4: Simulation settings used in the ASPEN PLUS real time optimizer input file ISMST.DAT.

Variable	Description	Units
STAB MOLEFLOW	STAB stream mole flow	lbmol/hr
STAB TEMP	STAB stream temperature	°F
STAB PRESS	STAB stream pressure	psia
STAB Z n	STAB stream composition ($n = 1 \ldots 23$)	lbmol/lbmol
C2REBL FLOW	C2 reboiler outlet mole flow	lbmol/hr
C2REBL TEMP	C2 reboiler outlet temperature	°F
C2REBL PRESS	C2 reboiler outlet pressure	psia
C2REBL Z n	C2 reboiler outlet composition ($n = 1 \ldots 23$)	lbmol/lbmol
E1 TEMP	E1 outlet temperature	°F
E1 PRESS	E1 outlet pressure	psia
C1 RFLX RATIO	C1 molar reflux ratio	lbmol/lbmol
C1 VAP/DIST	C1 condenser molar vapour fraction	lbmol/lbmol
C1 REBL FLOW	C1 reboiler outlet mole flow	lbmol/hr
C1 REBL TEMP	C1 reboiler outlet temperature	°F
C1 COL P DROP	C1 column pressure drop	psi
C1 TOP TRAY P	C1 condenser pressure	psia
C1 2ND TRAY P	C1 tray 2 pressure	psia
C2 DIST FLOW	LT-ISO stream mole flow	lbmol/hr
C2 COND SC T	C2 condenser subcooled temperature	°F
C2 COL P DROP	C2 column pressure drop	psi
C2 TOP TRAY P	C2 condenser pressure	psia
C2 2ND TRAY P	C2 tray 2 pressure	psia
C2-C4 IC FLOW	C4 feed stream mole flow	lbmol/hr
C2-C5 IC FLOW	C5 feed stream mole flow	lbmol/hr
C4 REBL DUTY	C4 reboiler heat duty	MMBTU/hr

continued on the next page ...

... Table B.4 continued from the previous page

Variable	Description	Units
C5 REBL DUTY	C5 reboiler heat duty	MMBTU/hr
C1 T, TRAY n	C1 temperature profile ($n = 1 \ldots 29$)	°F
C2 T, TRAY n	C2 temperature profile ($n = 1 \ldots 24$)	°F
C4 T, TRAY n	C4 temperature profile ($n = 1 \ldots 3$)	°F
C5 T, TRAY n	C5 temperature profile ($n = 1 \ldots 3$)	°F
C1 L, TRAY n	C1 liquid flow profile ($n = 1 \ldots 29$)	lbmol/hr
C1 V, TRAY n	C1 vapour flow profile ($n = 1 \ldots 29$)	lbmol/hr
C2 L, TRAY n	C2 liquid flow profile ($n = 1 \ldots 24$)	lbmol/hr
C2 V, TRAY n	C2 vapour flow profile ($n = 1 \ldots 24$)	lbmol/hr
C4 L, TRAY n	C4 liquid flow profile ($n = 1 \ldots 3$)	lbmol/hr
C4 V, TRAY n	C4 vapour flow profile ($n = 1 \ldots 3$)	lbmol/hr
C5 L, TRAY n	C5 liquid flow profile ($n = 1 \ldots 3$)	lbmol/hr
C5 V, TRAY n	C5 vapour flow profile ($n = 1 \ldots 3$)	lbmol/hr

B.6.7 RVARNC.DAT

`RVARNC.DAT`, contains standard deviations used to determine the normal distributions for the random noise to be added to the plant measurements. This input file has the the same format as `IVARNC.DAT`, with the 28 variances given in the same order as presented in Table B.2.

B.7 Results

Results from the real time optimizer are stored series of directories under the directory `results/` for each iteration, e.g., for the first iteration the results are stored in `results/itn1/`. The results files stored after each iteration are summarised in Table B.8.

Table B.5: Optimization parameters used in the ASPEN PLUS real time optimizer input file IOBJFN.DAT.

Variable	Description	Units
PRMRVP	Reid vapour pressure premium	$\$/m^3/kPa$
SPCRVP	Reid vapour pressure specification below which a quality premium is added to the LT-ISO stream price	kPa
DVAP	VAP stream price	$\$/m^3$
DDIST	DIST-C4 stream price	$\$/m^3$
DLTISO	LT-ISO stream price	$\$/m^3$
DHVISO	HV-ISO stream price	$\$/m^3$
DLTDST	LT-DIST stream price	$\$/m^3$
DSSFED	SSFEED stream price	$\$/m^3$
DSTAB	STAB feed stream price	$\$/m^3$
DFO	Fuel oil price	$\$/LFEm^3$
F1EFF	C1 reboiler furnace efficiency	—
F2EFF	C2 reboiler furnace efficiency	—

Table B.6: Constraint bounds used in the ASPEN PLUS real time optimizer input file ICNSTR.DAT.

Variable	Description	Units
A11U	DIST-C4 C5's mole fraction upper bound	lbmol/lbmol
A11L	DIST-C4 C5's mole fraction lower bound	lbmol/lbmol
A31UR	LT-ISO Reid vapour pressure upper bound	psi
A31LT	C2 condenser bubble point temperature lower bound	°F
A41U1	HV-ISO 10% TBP (true boiling point) temperature upper bound	°F
A41L1	HV-ISO 10% TBP temperature lower bound	°F

continued on the next page ...

... Table B.6 continued from the previous page

Variable	Description	Units
A41U9	HV-ISO 90% TBP temperature upper bound	°F
A41L9	HV-ISO 90% TBP temperature lower bound	°F
A51U1	LT-DIST 10% TBP temperature upper bound	°F
A51L1	LT-DIST 10% TBP temperature lower bound	°F
A51U9	LT-DIST 90% TBP temperature upper bound	°F
A51L9	LT-DIST 90% TBP temperature lower bound	°F
F12U	C1 reflux standard liquid volume flow upper bound	bbl/day
F12L	C1 reflux standard liquid volume flow lower bound	bbl/day
F32U	C2 reflux standard liquid volume flow upper bound	bbl/day
F41U	C4 feed stream standard liquid volume flow upper bound	bbl/day
F51U	C5 feed stream standard liquid volume flow upper bound	bbl/day
T11U	C1 tray 3 temperature upper bound	°F
T11L	C1 tray 3 temperature lower bound	°F
T22U	C1 reboiler outlet temperature upper bound	°F
T22L	C1 reboiler outlet temperature lower bound	°F
T31U	C2 tray 2 temperature upper bound	°F
T31L	C2 tray 2 temperature lower bound	°F
T62U	C2 reboiler outlet temperature upper bound	°F
T62L	C2 reboiler outlet temperature lower bound	°F

Table B.7: Optimization free variable ranges used in the ASPEN PLUS real time optimizer input file IOFREE.DAT.

Variable	Description	Units
C1-RR%	C1 molar reflux ratio	—
C1-RDV%	C1 condenser molar vapour fraction	—
C1-REBL-FLOW%	C1 reboiler mole flow	—
C1-REBL-TEMP	C1 reboiler outlet temperature	°F
E1-TEMP	E1 outlet temperature	°F
C2-MOLE-D%	C2 liquid distillate mole flow	—
C2-REBL-FLOW%	C2 reboiler mole flow	—
C2-REBL-TEMP	C2 reboiler outlet temperature	°F
C2-SC-TEMP	C2 condenser sub-cooled temperature	°F
C4-QN%	C4 reboiler heat duty	—
C5-QN%	C5 reboiler heat duty	—
C2-C4-FLOW%	C4 feed stream mole flow	—
C2-C5-FLOW%	C5 feed stream mole flow	—

Table B.8: ASEPN PLUS real time optimizer results files.

File	Description
RDVARS.DAT	Reconciled variables (in the same format as IMEAS.DAT)
RDOFF.DAT	Reconciled variable offsets (reconciled - measured)
RDERR.DAT	Data reconciliation error contributions for each variable
ROVARS.DAT	Predicted values of the measured variables after optimization
ROCNST.DAT	Inequality constraint results summary
ROOPFN.DAT	Optimization objective function results summary
ROSTPT.DAT	Set point results summary
ROSMST.DAT	ASPEN PLUS simulation settings after optimization
PLVARS.DAT	Measures variables from the plant simulation
IMEAS.DAT	Measured variables from the plant with noise added for the next iteration of the real time optimizer

B.8 Example

B.8.1 Input Files

The eight input files required to run an ASPEN PLUS real time optimization run are listed below. They have been printed in two columns to save paper, but appear in a single column in the actual input file.

IMEAS.DAT

```
F01     .20863014E+05
F11     .75209442E+05
F12     .70246257E+04
F13     .88440222E+03
F21     .31955347E+05
F22     .19080626E+05
F31     .15802698E+04
F32     .12862473E+05
F41     .50296623E+04
F42     .38176940E+04
F51     .32645036E+04
F52     .18477370E+04
F61     .29933726E+05
F62     .12250671E+05
T01     .41295470E+03
T12     .15233838E+03
T13     .96372107E+02
T21     .51601320E+03
T22     .58608488E+03
T23     .47969655E+03
T31     .12835124E+03
T32     .84722528E+02
T41     .24031157E+03
T42     .26107760E+03
T51     .35619866E+03
T52     .42389816E+03
T61     .62344743E+03
T62     .68389991E+03
```

IVARNC.DAT

```
VF01    .10431507E+04
VF11    .37604721E+04
```

```
VF12    .35123129E+03
VF13    .44220110E+02
VF21    .15977674E+04
VF22    .95403132E+03
VF31    .79013490E+02
VF32    .64312363E+03
VF41    .25148312E+03
VF42    .19088470E+03
VF51    .16322518E+03
VF52    .92386852E+02
VF61    .14966863E+04
VF62    .61253356E+03
VT01    .20647735E+02
VT12    .76269189E+01
VT13    .48186054E+01
VT21    .25800660E+02
VT22    .29304244E+02
VT23    .23984827E+02
VT31    .64175619E+01
VT32    .42361264E+01
VT41    .12015579E+02
VT42    .13053880E+02
VT51    .17809933E+02
VT52    .21194908E+02
VT61    .31172371E+02
VT62    .34194995E+02
```

IWGHT.DAT

```
WF01    .10000000E+01
WF11    .10000000E+01
WF12    .10000000E+01
WF13    .10000000E+01
WF21    .10000000E+01
WF22    .10000000E+01
WF31    .10000000E+01
WF32    .10000000E+01
WF41    .10000000E+01
WF42    .10000000E+01
WF51    .10000000E+01
WF52    .10000000E+01
WF61    .10000000E+01
WF62    .10000000E+01
WT01    .10000000E+01
WT12    .10000000E+01
```

```
        WT13     .10000000E+01
        WT21     .10000000E+01
        WT22     .10000000E+01
        WT23     .10000000E+01
        WT31     .10000000E+01
        WT32     .10000000E+01
        WT41     .10000000E+01
        WT42     .10000000E+01
        WT51     .10000000E+01
        WT52     .10000000E+01
        WT61     .10000000E+01
        WT62     .10000000E+01
```

IDFREE.DAT

```
STAB-MOLEFLOW    .10000000E+03
    STAB-TEMP    .30000000E+02
        C1-RR    .10000000E+01
       C1-RDV    .30000000E+00
 C1-REBL-FLOW    .40000000E+03
 C1-REBL-TEMP    .30000000E+02
      E1-TEMP    .20000000E+02
    C2-MOLE-D    .50000000E+02
 C2-REBL-FLOW    .20000000E+03
 C2-REBL-TEMP    .40000000E+02
   C2-SC-TEMP    .10000000E+02
        C4-QN    .15000000E+01
        C5-QN    .10000000E+01
   C2-C4-FLOW    .15000000E+03
   C2-C5-FLOW    .10000000E+03
```

ISMST.DAT

```
STAB MOLEFLOW    .14871771E+04
    STAB TEMP    .41295470E+03
   STAB PRESS    .20000000E+03
    STAB Z  1    .70368023E-03
    STAB Z  2    .24683008E-01
    STAB Z  3    .47364015E-01
    STAB Z  4    .38724012E-01
    STAB Z  5    .53531017E-01
    STAB Z  6    .31279010E-01
    STAB Z  7    .35193011E-01
    STAB Z  8    .16149005E-01
```

STAB Z 9	.47256015E-01
STAB Z 10	.43562014E-01
STAB Z 11	.49649016E-01
STAB Z 12	.53071017E-01
STAB Z 13	.52326017E-01
STAB Z 14	.48963016E-01
STAB Z 15	.44418014E-01
STAB Z 16	.41240013E-01
STAB Z 17	.60769019E-01
STAB Z 18	.53782017E-01
STAB Z 19	.59831019E-01
STAB Z 20	.62687020E-01
STAB Z 21	.56050018E-01
STAB Z 22	.52921017E-01
STAB Z 23	.25848008E-01
C2REBL FLOW	.13271950E+04
C2REBL TEMP	.68389991E+03
C2REBL PRESS	.35658100E+02
C2REBL Z 1	.00000000E+00
C2REBL Z 2	.22332348E-19
C2REBL Z 3	.37670678E-14
C2REBL Z 4	.34882008E-10
C2REBL Z 5	.23324335E-07
C2REBL Z 6	.52228058E-07
C2REBL Z 7	.10675699E-05
C2REBL Z 8	.67296430E-06
C2REBL Z 9	.53528080E-05
C2REBL Z 10	.13837890E-04
C2REBL Z 11	.42296535E-04
C2REBL Z 12	.12371797E-03
C2REBL Z 13	.36380606E-03
C2REBL Z 14	.11700160E-02
C2REBL Z 15	.45504148E-02
C2REBL Z 16	.18704296E-01
C2REBL Z 17	.10406397E+00
C2REBL Z 18	.14438669E+00
C2REBL Z 19	.16838694E+00
C2REBL Z 20	.17711363E+00
C2REBL Z 21	.15842350E+00
C2REBL Z 22	.14958787E+00
C2REBL Z 23	.73061840E-01
E1 TEMP	.47965650E+03
E1 PRESS	.34000000E+02
C1 RFLX RATIO	.31563083E+01
C1 VAP/DIST	.60262460E+00

C1 REBL FLOW	.20056994E+04
C1 REBL TEMP	.58608488E+03
C1 COL P DROP	.60000000E+01
C1 TOP TRAY P	.16300000E+03
C1 2ND TRAY P	.16700000E+03
C2 DIST FLOW	.18657490E+03
C2 COND SC T	.84722528E+02
C2 COL P DROP	.12700000E+02
C2 TOP TRAY P	.18958100E+02
C2 2ND TRAY P	.22958100E+02
C2-C4 IC FLOW	.47761110E+03
C2-C5 IC FLOW	.24821130E+03
C4 REBL DUTY	.19009000E+01
C5 REBL DUTY	.23992000E+01
C1 T, TRAY 1	.89535345E+02
C1 T, TRAY 2	.14904247E+03
C1 T, TRAY 3	.16925629E+03
C1 T, TRAY 4	.18204352E+03
C1 T, TRAY 5	.19214284E+03
C1 T, TRAY 6	.20025927E+03
C1 T, TRAY 7	.20663956E+03
C1 T, TRAY 8	.21175995E+03
C1 T, TRAY 9	.21638377E+03
C1 T, TRAY 10	.22163349E+03
C1 T, TRAY 11	.22935183E+03
C1 T, TRAY 12	.24337344E+03
C1 T, TRAY 13	.27515930E+03
C1 T, TRAY 14	.38666948E+03
C1 T, TRAY 15	.38944282E+03
C1 T, TRAY 16	.39111377E+03
C1 T, TRAY 17	.39215763E+03
C1 T, TRAY 18	.39289114E+03
C1 T, TRAY 19	.39348955E+03
C1 T, TRAY 20	.39405805E+03
C1 T, TRAY 21	.39467415E+03
C1 T, TRAY 22	.39541209E+03
C1 T, TRAY 23	.39636461E+03
C1 T, TRAY 24	.39768443E+03
C1 T, TRAY 25	.39970686E+03
C1 T, TRAY 26	.40337775E+03
C1 T, TRAY 27	.41184885E+03
C1 T, TRAY 28	.43649879E+03
C1 T, TRAY 29	.51983431E+03
C2 T, TRAY 1	.11276069E+03
C2 T, TRAY 2	.15092922E+03

```
C2 T, TRAY  3     .16942048E+03
C2 T, TRAY  4     .18113435E+03
C2 T, TRAY  5     .18875257E+03
C2 T, TRAY  6     .19400527E+03
C2 T, TRAY  7     .19799118E+03
C2 T, TRAY  8     .20144500E+03
C2 T, TRAY  9     .20492805E+03
C2 T, TRAY 10     .20897115E+03
C2 T, TRAY 11     .21419846E+03
C2 T, TRAY 12     .22148888E+03
C2 T, TRAY 13     .23233251E+03
C2 T, TRAY 14     .24964309E+03
C2 T, TRAY 15     .27801566E+03
C2 T, TRAY 16     .32073101E+03
C2 T, TRAY 17     .35897822E+03
C2 T, TRAY 18     .38821133E+03
C2 T, TRAY 19     .41585501E+03
C2 T, TRAY 20     .44186541E+03
C2 T, TRAY 21     .48339188E+03
C2 T, TRAY 22     .53118577E+03
C2 T, TRAY 23     .57017714E+03
C2 T, TRAY 24     .62644190E+03
C4 T, TRAY  1     .27879455E+03
C4 T, TRAY  2     .28219693E+03
C4 T, TRAY  3     .29203449E+03
C5 T, TRAY  1     .40277854E+03
C5 T, TRAY  2     .41487987E+03
C5 T, TRAY  3     .42950162E+03
C1 L, TRAY  1     .10220247E+04
C1 L, TRAY  2     .99126282E+03
C1 L, TRAY  3     .98907993E+03
C1 L, TRAY  4     .97447261E+03
C1 L, TRAY  5     .96286239E+03
C1 L, TRAY  6     .95553052E+03
C1 L, TRAY  7     .94985039E+03
C1 L, TRAY  8     .94225902E+03
C1 L, TRAY  9     .92830425E+03
C1 L, TRAY 10     .90091906E+03
C1 L, TRAY 11     .84560519E+03
C1 L, TRAY 12     .72395584E+03
C1 L, TRAY 13     .41396661E+03
C1 L, TRAY 14     .14499493E+04
C1 L, TRAY 15     .15008605E+04
C1 L, TRAY 16     .15314827E+04
C1 L, TRAY 17     .15503670E+04
```

```
C1 L, TRAY 18    .15633516E+04
C1 L, TRAY 19    .15736414E+04
C1 L, TRAY 20    .15830949E+04
C1 L, TRAY 21    .15930030E+04
C1 L, TRAY 22    .16045027E+04
C1 L, TRAY 23    .16188608E+04
C1 L, TRAY 24    .16378243E+04
C1 L, TRAY 25    .16643391E+04
C1 L, TRAY 26    .17039787E+04
C1 L, TRAY 27    .17644646E+04
C1 L, TRAY 28    .18189029E+04
C1 L, TRAY 29    .32052806E+04
C1 V, TRAY  1    .17331234E+03
C1 V, TRAY  2    .11953371E+04
C1 V, TRAY  3    .12788587E+04
C1 V, TRAY  4    .12766758E+04
C1 V, TRAY  5    .12620685E+04
C1 V, TRAY  6    .12504583E+04
C1 V, TRAY  7    .12431264E+04
C1 V, TRAY  8    .12374463E+04
C1 V, TRAY  9    .12298549E+04
C1 V, TRAY 10    .12159001E+04
C1 V, TRAY 11    .11885149E+04
C1 V, TRAY 12    .11332011E+04
C1 V, TRAY 13    .10115517E+04
C1 V, TRAY 14    .70156248E+03
C1 V, TRAY 15    .25036802E+03
C1 V, TRAY 16    .30127931E+03
C1 V, TRAY 17    .33190147E+03
C1 V, TRAY 18    .35078573E+03
C1 V, TRAY 19    .36377042E+03
C1 V, TRAY 20    .37406012E+03
C1 V, TRAY 21    .38351364E+03
C1 V, TRAY 22    .39342177E+03
C1 V, TRAY 23    .40492142E+03
C1 V, TRAY 24    .41927954E+03
C1 V, TRAY 25    .43824309E+03
C1 V, TRAY 26    .46475786E+03
C1 V, TRAY 27    .50439749E+03
C1 V, TRAY 28    .56488337E+03
C1 V, TRAY 29    .61932164E+03
C2 L, TRAY  1    .27761287E+04
C2 L, TRAY  2    .29082599E+04
C2 L, TRAY  3    .29094240E+04
C2 L, TRAY  4    .29151352E+04
```

```
C2 L, TRAY  5    .29199472E+04
C2 L, TRAY  6    .29222287E+04
C2 L, TRAY  7    .29208497E+04
C2 L, TRAY  8    .29141923E+04
C2 L, TRAY  9    .28996378E+04
C2 L, TRAY 10    .28730864E+04
C2 L, TRAY 11    .28279887E+04
C2 L, TRAY 12    .27524899E+04
C2 L, TRAY 13    .26232345E+04
C2 L, TRAY 14    .24106877E+04
C2 L, TRAY 15    .21431812E+04
C2 L, TRAY 16    .14967844E+04
C2 L, TRAY 17    .14080986E+04
C2 L, TRAY 18    .13224056E+04
C2 L, TRAY 19    .97377810E+03
C2 L, TRAY 20    .76594586E+03
C2 L, TRAY 21    .12985384E+04
C2 L, TRAY 22    .14919878E+04
C2 L, TRAY 23    .14798866E+04
C2 L, TRAY 24    .18533342E+04
C2 V, TRAY  1    .00000000E+00
C2 V, TRAY  2    .27761287E+04
C2 V, TRAY  3    .30948348E+04
C2 V, TRAY  4    .30959989E+04
C2 V, TRAY  5    .31017101E+04
C2 V, TRAY  6    .31065221E+04
C2 V, TRAY  7    .31088036E+04
C2 V, TRAY  8    .31074246E+04
C2 V, TRAY  9    .31007672E+04
C2 V, TRAY 10    .30862127E+04
C2 V, TRAY 11    .30596613E+04
C2 V, TRAY 12    .30145636E+04
C2 V, TRAY 13    .29390648E+04
C2 V, TRAY 14    .28098094E+04
C2 V, TRAY 15    .24735242E+04
C2 V, TRAY 16    .22060176E+04
C2 V, TRAY 17    .20372320E+04
C2 V, TRAY 18    .18333293E+04
C2 V, TRAY 19    .17476363E+04
C2 V, TRAY 20    .16472202E+04
C2 V, TRAY 21    .14393879E+04
C2 V, TRAY 22    .77239922E+03
C2 V, TRAY 23    .96584867E+03
C2 V, TRAY 24    .95374742E+03
C4 L, TRAY  1    .47655706E+03
```

```
C4 L, TRAY  2     .47723223E+03
C4 L, TRAY  3     .35387266E+03
C4 V, TRAY  1     .12373844E+03
C4 V, TRAY  2     .12268440E+03
C4 V, TRAY  3     .12335957E+03
C5 L, TRAY  1     .25855310E+03
C5 L, TRAY  2     .26335025E+03
C5 L, TRAY  3     .13299452E+03
C5 V, TRAY  1     .11521678E+03
C5 V, TRAY  2     .12555859E+03
C5 V, TRAY  3     .13035573E+03
```

IOBJFN.DAT

```
      PRMRVP    -.18000000E+00
      SPCRVP     .97000000E+02
        DVAP     .15000000E+00
       DDIST     .10007000E+03
      DLTISO     .18269000E+03
      DHVISO     .19389000E+03
      DLTDST     .18052000E+03
      DSSFED     .15834000E+03
       DSTAB     .15834000E+03
         DFO     .82140000E+02
       F1EFF     .70000000E+00
       F2EFF     .70000000E+00
```

ICNSTR.DAT

```
        A11U     .15000000E+00
        A11L     .50000000E-02
       A31UR     .14000000E+02
       A31LT     .11000000E+03
       A41U1     .26600000E+03
       A41L1     .17600000E+03
       A41U9     .46400000E+03
       A41L9     .28400000E+03
       A51U1     .55400000E+03
       A51L1     .30200000E+03
       A51U9     .55400000E+03
       A51L9     .30200000E+03
        F12U     .90000000E+04
        F12L     .10000000E+04
        F32U     .25000000E+05
```

F41U	.10604000E+05
F51U	.60000000E+04
T11U	.22000000E+03
T11L	.12500000E+03
T22U	.67500000E+03
T22L	.53000000E+03
T31U	.25000000E+03
T31L	.12000000E+03
T62U	.70000000E+03
T62L	.55000000E+03

IOFREE.DAT

C1-RR%	.50000000E-01
C1-RDV%	.50000000E-01
C1-REBL-FLOW%	.50000000E-01
C1-REBL-TEMP	.50000000E+01
E1-TEMP	.50000000E+01
C2-MOLE-D%	.50000000E-01
C2-REBL-FLOW%	.50000000E-01
C2-REBL-TEMP	.50000000E+01
C2-SC-TEMP	.50000000E+01
C4-QN%	.50000000E-01
C5-QN%	.50000000E-01
C2-C4-FLOW%	.50000000E-01
C2-C5-FLOW%	.50000000E-01

RVARNC.DAT

VF01	.10431507E+04
VF11	.37604721E+04
VF12	.35123129E+03
VF13	.44220110E+02
VF21	.15977674E+04
VF22	.95403132E+03
VF31	.79013490E+02
VF32	.64312363E+03
VF41	.25148312E+03
VF42	.19088470E+03
VF51	.16322518E+03
VF52	.92386852E+02
VF61	.14966863E+04
VF62	.61253356E+03
VT01	.20647735E+02

```
VT12    .76269189E+01
VT13    .48186054E+01
VT21    .25800660E+02
VT22    .29304244E+02
VT23    .23984827E+02
VT31    .64175619E+01
VT32    .42361264E+01
VT41    .12015579E+02
VT42    .13053880E+02
VT51    .17809933E+02
VT52    .21194908E+02
VT61    .31172371E+02
VT62    .34194995E+02
```

B.8.2 Command Line

If it the ASPEN PLUS real time optimizer has not already been setup, then type the command:

```
setup
```

When completed, type the command:

```
nohup nice run 1 &
```

to run one cycle of the real time optimizer.

B.8.3 Results

The following results will be present in the `results/itn1/` directory.

RDVARS.DAT

```
F01     .21859091E+05
F11     .69854292E+05
F12     .70514797E+04
F13     .82877968E+03
F21     .31262989E+05
F22     .19958397E+05
F31     .14784074E+04
F32     .14148616E+05
```

```
F41     .5577i892E+04
F42     .38080990E+04
F51     .33253764E+04
F52     .18961467E+04
F61     .27970047E+05
F62     .12775711E+05
T01     .43103635E+03
T12     .15716275E+03
T13     .94728241E+02
T21     .52429749E+03
T22     .59027415E+03
T23     .46081466E+03
T31     .14812960E+03
T32     .82876202E+02
T41     .26395977E+03
T42     .28408165E+03
T51     .36881493E+03
T52     .42521338E+03
T61     .60180580E+03
T62     .64621622E+03
```

RDOFF.DAT

```
OFFF01     .99607672E+03
OFFF11   -.53551503E+04
OFFF12     .26854007E+02
OFFF13   -.55622538E+02
OFFF21   -.69235822E+03
OFFF22     .87777126E+03
OFFF31   -.10186238E+03
OFFF32     .12861433E+04
OFFF41     .54752691E+03
OFFF42   -.95949819E+01
OFFF51     .60872806E+02
OFFF52     .48409696E+02
OFFF61   -.19636790E+04
OFFF62     .52504041E+03
OFFT01     .18081649E+02
OFFT12     .48243656E+01
OFFT13   -.16438662E+01
OFFT21     .82842879E+01
OFFT22     .41892701E+01
OFFT23   -.18881893E+02
OFFT31     .19778362E+02
OFFT32   -.18463257E+01
```

```
OFFT41     .23648201E+02
OFFT42     .23004048E+02
OFFT51     .12616267E+02
OFFT52     .13152236E+01
OFFT61    -.21641631E+02
OFFT62    -.37683687E+02
```

RDERR.DAT

```
ERRF01     .91178296E+00
ERRF11     .20279563E+01
ERRF12     .58456362E-02
ERRF13     .15822021E+01
ERRF21     .18777363E+00
ERRF22     .84652043E+00
ERRF31     .16619775E+01
ERRF32     .39993532E+01
ERRF41     .47401629E+01
ERRF42     .25266558E-02
ERRF51     .13908245E+00
ERRF52     .27456449E+00
ERRF61     .17213905E+01
ERRF62     .73472647E+00
ERRT01     .76688675E+00
ERRT12     .40011255E+00
ERRT13     .11638317E+00
ERRT21     .10309764E+00
ERRT22     .20436931E-01
ERRT23     .61975195E+00
ERRT31     .94981828E+01
ERRT32     .18996729E+00
ERRT41     .38735289E+01
ERRT42     .31054840E+01
ERRT51     .50180755E+00
ERRT52     .38506686E-02
ERRT61     .48199253E+00
ERRT62     .12144557E+01
```

ROVARS.DAT

```
F01     .21859091E+05
F11     .67107186E+05
F12     .69137124E+04
F13     .90205497E+03
```

F21	.30135826E+05
F22	.19932133E+05
F31	.15596100E+04
F32	.13951685E+05
F41	.59982861E+04
F42	.42020112E+04
F51	.33344776E+04
F52	.19524427E+04
F61	.29866890E+05
F62	.12218003E+05
T01	.43103635E+03
T12	.15824029E+03
T13	.93358741E+02
T21	.52629028E+03
T22	.59510129E+03
T23	.46581466E+03
T31	.15302258E+03
T32	.87876202E+02
T41	.27332984E+03
T42	.29468224E+03
T51	.38669777E+03
T52	.45288660E+03
T61	.61140730E+03
T62	.65121622E+03

ROOPFN.DAT

OBFUN-PROFIT	-.98374539E+01
TOT PROD VALS	.23035020E+05
TOT FEED COST	.22928410E+05
TOT UTIL COST	.11644748E+03
VAP SVOLFLOW	.10249031E+04
DISTC4 SVOLFL	.90205497E+03
LTISO SVOLFL	.15596100E+04
HVISO SVOLFL	.42020112E+04
LTDIST SVOLFL	.19524427E+04
SSFEED SVOLFL	.12218003E+05
STAB SVOLFLOW	.21859091E+05
C1 REBL DUTY	.21242019E+02
C2 REBL DUTY	.18081425E+02
LT-ISO MARG	.18277511E+03
LT-ISO RVP	.13999996E+02

ROCNST.DAT

```
          F12     .69137124E+04
          F32     .13951685E+05
          F41     .59982861E+04
          F51     .33344776E+04
          T11     .18125698E+03
          T22     .59510129E+03
          T31     .15302258E+03
          T62     .65121622E+03
   A11 (C5S)      .11881676E+00
   A31 (RVP)      .13999996E+02
  A31 (TBUB)      .11505224E+03
 A41 (TBP10)      .19737562E+03
 A41 (TBP90)      .35162453E+03
 A51 (TBP10)      .33825622E+03
 A51 (TBP90)      .49624756E+03
```

ROSTPT.DAT

```
          C1 REFLUX FLOW (F12) [BBL/DAY]     .69137124E+04
        C1 TRAY 3 TEMPERATURE (T11) [F]      .18125698E+03
          DIST-C4 C5 MOLE FRACTION (A11)     .11881676E+00
      C1 REBOILER TEMPERATURE (T22) [F]      .59510129E+03
          C2 REFLUX FLOW (F32) [BBL/DAY]     .13951685E+05
            C4 FEED FLOW (F41) [BBL/DAY]     .59982861E+04
            C5 FEED FLOW (F51) [BBL/DAY]     .33344776E+04
      C2 REBOILER TEMPERATURE (T62) [F]      .65121622E+03
   HV-ISO 10% TBP TEMPERATURE (A41) [F]      .19737562E+03
   HV-ISO 90% TBP TEMPERATURE (A41) [F]      .35162453E+03
  LT-DIST 10% TBP TEMPERATURE (A51) [F]      .33825622E+03
  LT-DIST 90% TBP TEMPERATURE (A51) [F]      .49624756E+03
```

Appendix C

SPEEDUP Real Time Optimizer Documentation

C.1 Purpose

This program performs a given number of cycles of a real time optimizer on the stablizer-splitter flowsheet given in Figure C.1 using SPEEDUP to perform the data reconciliation and optimization.

C.2 Specification

To setup the real time optimizer for a run, use the command:

```
setup
```

or

```
nohup setup &
```

This command must be given before running the real time optimizer, and the files listed in Table C.1 must already exist on the paths specified.

To run the real time optimizer for a given number of cycles, n, use the command:

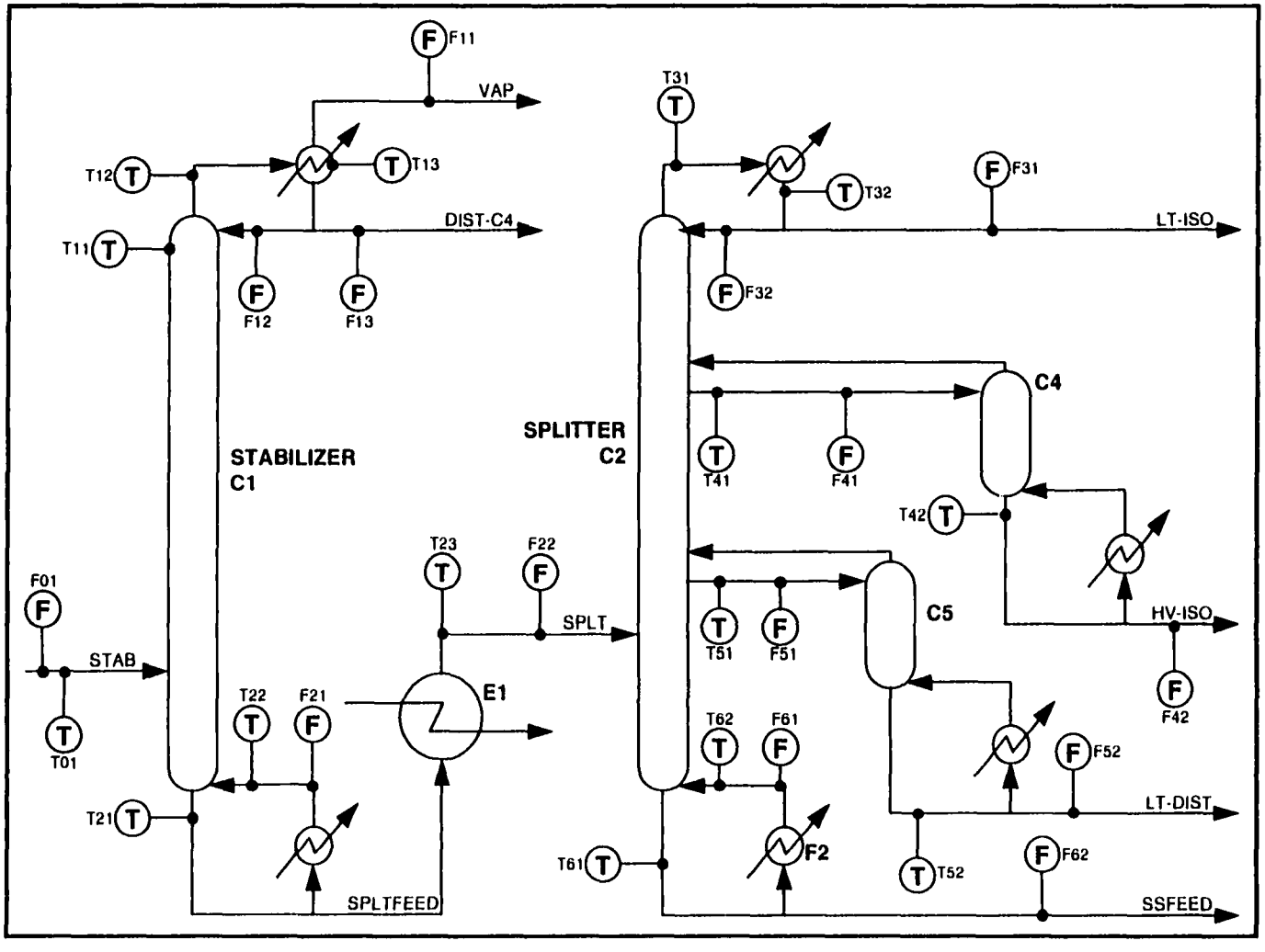


Figure C.1: Stabilizer-splitter process flowsheet (including measurement tag numbers).

`run` n

or

`nohup run` n `&`

The real time optimizer must already have been either `setup` or `run` prior to using this command.

C.3 Description

The SPEEDUP real time optimizer performs data reconciliation and optimization on the stabilizer-splitter flowsheet given in Figure C.1. Detailed information about this optimizer is covered by Naysmith [1].

Before running the real time optimizer, the program `setup` must be run first. This program translates and compiles the SPEEDUP and ASPEN PLUS simulations, and compiles the FORTRAN77 routines used. The files listed in Table C.1 must exist on the paths specified, and the NAG FORTRAN Library Routines must be available. The program `setup` only needs to be run once.

The program `run` runs a specified number of cycles (or iterations) of the real time optimizer. The steps used by this program are summarised in Figure C.2.

The data reconciliation uses a weighted least squares objective function to reconcile 28 measured variables (all variables shown in Figure C.1 except T11):

$$\min_{\mathbf{x}} \left[\sum_{i=1}^{28} w_i \left(\frac{y_i - x_i}{s_i} \right)^2 \right] \qquad \text{(C.1)}$$
$$\text{subject to } h(\mathbf{x}) = \mathbf{0}$$

where w = weighting factor associated with each measured variable;
y = the measured variable;
x = the estimated true (reconciled) variable value;

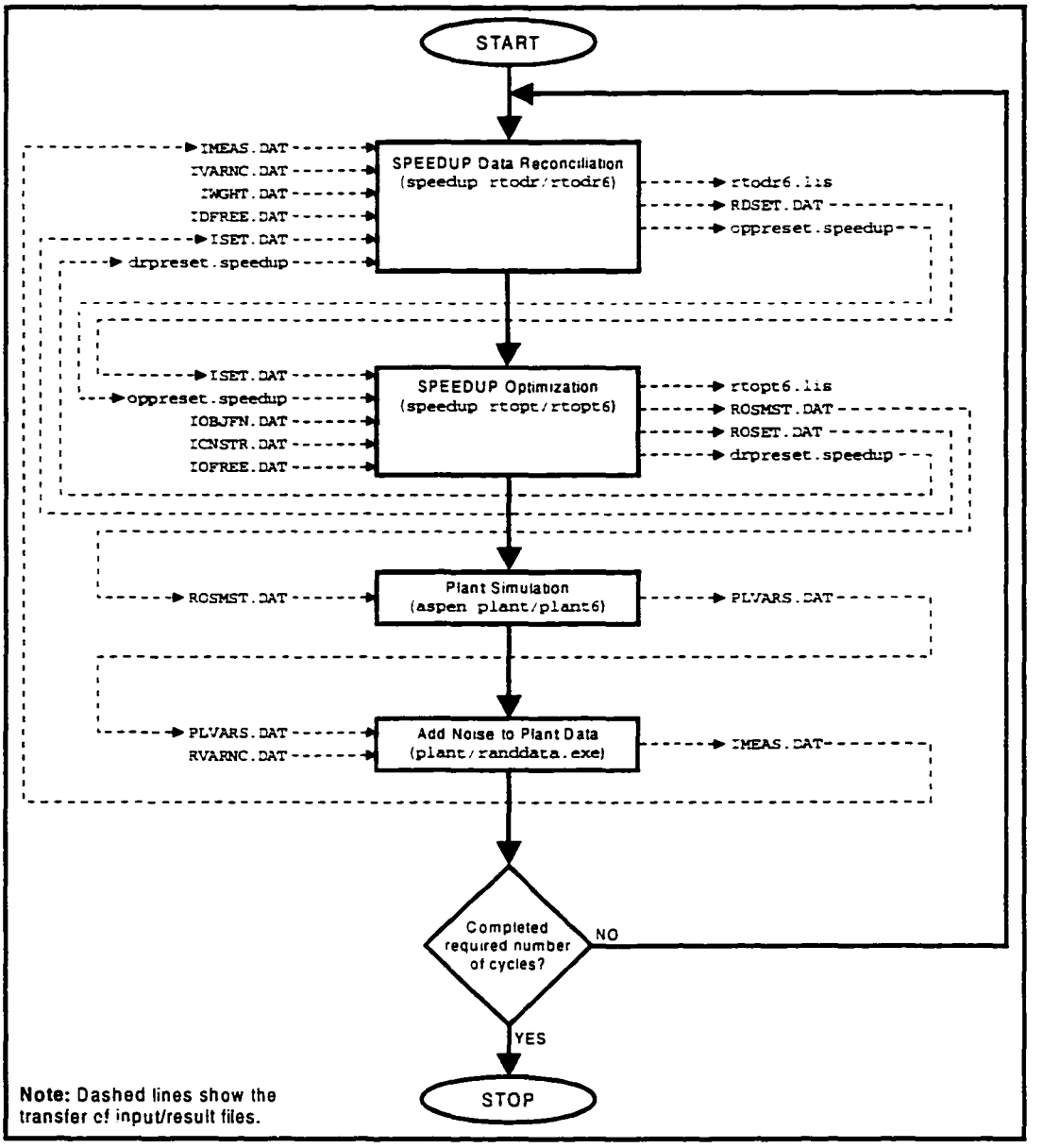


Figure C.2: SPEEDUP real time optimization program flow chart.

s = the standard deviation of the measurements;
$h(\mathbf{x})$ = the set of equality constraints representing the process model.

The optimization algorithm used is the SRQP (sequential reduced quadratic programming) algorithm using a line search method with a modified Lagrangian function. This objective function is minimized by varying 15 free variables.

The economic optimization maximizes profit using the following objective function:

$$\begin{aligned} &\max \left[\sum(\text{product values}) - (\text{feed cost}) - (\text{reboiler fuel oil cost})\right] \\ &\text{subject to } h(\mathbf{x}) = \mathbf{0} \\ &\qquad\qquad g(\mathbf{x}) \geq \mathbf{0} \end{aligned} \tag{C.2}$$

where $h(\mathbf{x})$ = the set of equality constraints representing the process model;
$g(\mathbf{x})$ = product quality and operating inequality constraints.

The optimization algorithm used is the SRQP algorithm using a line search method with a modifies augmented Lagrangian function. The objective function is maximized by varying 11 free variables subject to 25 inequality constraints.

Normally distributed random noise is added to the plant data calculated in the plant simulation before being returned to the real time optimizer for the next iteration. The noise has a mean of zero, and a given standard deviation for each of the 28 measured variables.

C.4 References

[1] M. Naysmith. Real time optimization of chemical processes — a comparison of closed versus open form equations using commercial simulation packages. Ph. D. Thesis, University of Waterloo, 1997.

C.5 Files

The files required to run SPEEDUP real time optimizer are listed in Table C.1.

C.6 Input File Specifications

All input files are formatted in the same way, except `ISET.DAT`, using the following FORTRAN format statement:

```
FORMAT(A13, 2X, E15.8)
```

The first 13 columns of each line contain a descriptive label. The next 2 columns are blank and the last 15 columns contain the corresponding numerical value. `ISET.DAT` is formatted using the following FORTRAN format statement:

```
FORMAT(A40, 2X, E15.8)
```

In this case, first the 40 columns of each line contain a descriptive label, the next 2 columns are blank and the last 15 columns contain the corresponding numerical value. Example input files are given in section C.8.

C.6.1 IMEAS.DAT, IVARNC.DAT and IWGHT.DAT

`IMEAS.DAT`, `IVARNC.DAT` and `IWGHT.DAT` contain the measured values, standard deviations and weights for data reconciliation corresponding to the 28 measured plant variables respectively. The 28 variables are given in the same order as presented in Table C.2 for all three files. The units given are for the measured variables and standard deviations. The variable names are the same as those used in Figure C.1.

Table C.1: SPEEDUP real time optimizer files required.

File	Description
`run`	Runs the real time optimizer (C-shell script, executable)
`setup`	Sets up the real time optimizer prior to being run (C-shell script, executable)
`plant/plant6.inp`	Steady state plant simulation (ASPEN PLUS input file)
`plant/randata.f`	Adds normally distributed random noise to the measured plant data (FORTRAN77 input file, requires the NAG FORTRAN Library Routines)
`plant/RVARNC.DAT`	Contains the random noise standard deviations for the measured plant data
`props/prop6.inp`	Properties for both the data reconciliation and optimization SPEEDUP simulations (PROPERTIES PLUS input file)
`rtodr/drpreset.speedup`	Results from a previous run used to initialize the data reconciliation simulation (SPEEDUP saved results file)
`rtodr/IDFREE.DAT`	Contains the data reconciliation free variable bound ranges
`rtodr/IMEAS.DAT`	Contains the measured variable data from the plant (with noise added)
`rtodr/ISET.DAT`	Contains the initial simulation settings and parameters for starting the SPEEDUP real time optimization simulations
`rtodr/IVARNC.DAT`	Contains the measured variable standard deviations for the data reconciliation
`rtodr/IWGHT.DAT`	Contains the measured variable weights for the data reconciliation

continued on the next page ...

... Table C.1 continued from the previous page

File	Description
rtodr/ model6x.speedup	User written models used in the process simulation flowsheet (SPEEDUP input file)
rtodr/ report6y.speedup	User written reports (SPEEDUP input file)
rtodr/rtodr6.f	EDI (external data interface) code for receiving and transmitting variables (FORTRAN input file)
rtodr/ rtodr6.speedup	Data reconciliation simulation (SPEEDUP input file)
rtodr/ rtodr6run.cmd	SPEEDUP commands used during batch run of the data reconciliation simulation
rtodr/ rtodr6setup.cmd	SPEEDUP commands used during setup of the data reconciliation simulation
rtodr/ speedup.include	File names of FORTRAN object code to be included in the SPEEDUP simulation when compiling
rtopt/ICNSTR.DAT	Contains the optimization inequality constraint upper and lower bounds
rtopt/IOBJFN.DAT	Contains the optimization objective function parameters
rtopt/IOFREE.DAT	Contains the optimization free variable bound ranges
rtopt/ model6x.speedup	User written models used in the process simulation flowsheet (SPEEDUP input file)
rtopt/ report6y.speedup	User written reports (SPEEDUP input file)
rtopt/rtopt6.f	EDI code for receiving and transmitting variables (FORTRAN input file)

continued on the next page ...

... Table C.1 continued from the previous page

File	Description
`rtopt/` `rtopt6.speedup`	Optimization simulation (SPEEDUP input file)
`rtopt/` `rtopt6run.cmd`	SPEEDUP commands used during batch run of the optimization simulation
`rtopt/` `rtopt6setup.cmd`	SPEEDUP commands used during setup of the optimization simulation
`rtopt/` `speedup.include`	File names of FORTRAN object code to be included in the SPEEDUP simulation when compiling

C.6.2 IDFREE.DAT

`IDFREE.DAT` contains the free variable ranges for data reconciliation. The upper and lower are free variable bounds are calculated using the following formulae:

$$\begin{aligned} x_{UB} &= x^{(0)} + r \\ x_{LB} &= x^{(0)} - r \end{aligned} \tag{C.3}$$

where x_{UB} = free variable upper bound;
x_{UB} = free variable lower bound;
$x^{(0)}$ = initial free variable value;
r = free variable range as specified in `IDFREE.DAT`.

The free variable range specifications are given in the order shown in Table C.3.

C.6.3 ISET.DAT

The file `ISET.DAT` contains all the simulation settings to initialize the SPEEDUP real time optimization simulation. Table C.4 summarises the variable in this file. Variables which are vectors, such as compositions, use n lines in the

Table C.2: Variable order used in the SPEEDUP real time optimizer input files `IMEAS.DAT`, `IVARNC.DAT` and `IWGHT.DAT`.

Variable	Description	Units
F01	STAB stream standard liquid volume flow	bbl/day
F11	VAP stream standard vapour volume flow	cuft/hr
F12	C1 reflux standard liquid volume flow	bbl/day
F13	DIST-C4 stream standard liquid volume flow	bbl/day
F21	C1 reboiler standard liquid volume flow	bbl/day
F22	E1 outlet standard liquid volume flow	bbl/day
F31	LT-ISO stream standard liquid volume flow	bbl/day
F32	C2 reflux standard liquid volume flow	bbl/day
F41	C4 feed standard liquid volume flow	bbl/day
F42	HV-ISO stream standard liquid volume flow	bbl/day
F51	C5 feed standard liquid volume flow	bbl/day
F52	LT-DIST stream standard liquid volume flow	bbl/day
F61	C2 reboiler standard liquid volume flow	bbl/day
F62	SSFEED stream standard liquid volume flow	bbl/day
T01	STAB stream temperature	°F
T12	C1 tray 2 temperature	°F
T13	C1 condenser temperature	°F
T21	C1 bottoms stream temperature	°F
T22	C1 reboiler outlet temperature	°F
T23	E1 outlet temperature	°F
T31	C2 tray 2 temperature	°F
T32	LT-ISO stream temperature	°F
T41	C4 feed stream temperature	°F
T42	HV-ISO stream temperature	°F
T51	C5 feed stream temperature	°F
T52	LT-DIST stream temperature	°F
T61	C2 bottoms stream temperature	°F
T62	C2 reboiler outlet temperature	°F

Table C.3: Data reconciliation free variable ranges used in the SPEEDUP real time optimizer input file IDFREE.DAT.

Variable	Description	Units
C1COND.Q	C1 condenser heat duty	MMBTU/hr
C1SPL1.L_OUT	DIST-C4 stream mole flow	lbmol/hr
C1FT14.F_IN	STAB stream mole flow	lbmol/hr
C1FT14.T_IN	STAB stream temperature	°F
C1BT29.L_OUT1	E1 inlet mole flow	lbmol/hr
C1BT29.L_OUT2	C1 reboiler mole flow	lbmol/hr
C2COND.Q	C2 condenser heat duty	MMBTU/hr
C2COND.T_DROP	C2 condenser subcooled temperature drop (bubble point less subcooled temperature)	°F
C2FT21.T_IN	E1 outlet temperature	°F
C2BT24.L_OUT1	SSFEED stream mole flow	lbmol/hr
C2BT24.L_OUT2	C2 reboiler mole flow	lbmol/hr
C2SPL1.L_OUT1	C4 feed stream mole flow	lbmol/hr
C2SPL2.L_OUT1	C5 feed stream mole flow	lbmol/hr
C4REBL.L_OUT	HV-ISO stream mole flow	lbmol/hr
C5REBL.L_OUT	LT-DIST stream mole flow	lbmol/hr

input file as specified in Table C.4. Note that this file is formatted differently to the other input files using the FORTRAN format statement:

```
FORMAT(A40, 2X, E15.8)
```

C.6.4 IOBJFN.DAT

`IOBJFN.DAT` contains the parameters used in calculating the economic objective function. Table C.5 summarises the order in which these parameters must appear in the file `IOBJFN.DAT`.

C.6.5 ICNSTR.DAT

`ICNSTR.DAT` contains the upper and lower bounds for all the inequality constraints in the optimization. Table C.6 summarises the order in which these bounds must appear in the file `ICNSTR.DAT`.

C.6.6 IOFREE.DAT

`IOFREE.DAT` contains the free variable ranges for optimization. The upper and lower are free variable bounds are calculated using the following formulae for all free variables except temperatures:

$$\begin{aligned} x_{UB} &= x^{(0)} + qx^{(0)} \\ x_{LB} &= x^{(0)} - qx^{(0)} \end{aligned} \tag{C.4}$$

where $q =$ free variable range as a fraction of the initial variable value.

Temperature free variable bounds are calculated using the formulae given in equation (C.3). The free variable range specifications are given in the order shown in Table C.7. Free variable bounds calculated using equation (C.4) have the character '%' at the end of their variable name.

Table C.4: Simulation settings used in the SPEEDUP real time optimizer input file ISET.DAT.

Variable	Description	Units
MFEED.Z_OUT(n)	STAB stream composition ($n = 1 \ldots 22$) (Note: the 23rd component composition, B395T510, is not included)	lbmol/lbmol
MFEED.P_OUT	STAB stream pressure	psia
C1COND.P	C1 condenser pressure	psia
C1COND.Q	C1 condenser heat duty	MMBTU/hr
C1SPL1.L_OUT1	DIST-C4 stream mole flow	lbmol/hr
"C1S.TRAY_SS(2)".P	C1 tray 2 pressure	psia
C1FT14.F_IN	STAB stream mole flow	lbmol/hr
C1FT14.T_IN	STAB stream temperature	°F
C1BT29.L_OUT1	E1 inlet mole flow	lbmol/hr
C1BT29.L_OUT2	C1 reboiler mole flow	lbmol/hr
E1.P_OUT	E1 outlet pressure	psia
C2COND.P	C2 condenser pressure	psia
C2COND.Q	C2 condenser heat duty	MMBTU/hr
C2COND.T_DROP	C2 condenser subcooled temperature drop	°F
"C2S1.TRAY_SS(2)".P	C2 tray 2 pressure	psia
C2FT21.T_IN	E1 outlet temperature	°F
C2BT24.L_OUT1	SSFEED stream mole flow	lbmol/hr
C2BT24.L_OUT2	C2 reboiler mole flow	lbmol/hr
C2SPL1.L_OUT1	C4 feed stream mole flow	lbmol/hr
C2SPL2.L_OUT1	C5 feed stream mole flow	lbmol/hr
C4REBL.L_OUT	HV-ISO stream mole flow	lbmol/hr
C5REBL.L_OUT	LT-DIST stream mole flow	lbmol/hr
F31.TL_OUT	LT-ISO stream temperature	°F

Table C.5: Optimization parameters used in the SPEEDUP real time optimizer input file IOBJFN.DAT.

Variable	Description	Units
PRMRVP	Reid vapour pressure premium	$\$/m^3/kPa$
SPCRVP	Reid vapour pressure specification below which a quality premium is added to the LT-ISO stream price	kPa
DVAP	VAP stream price	$\$/m^3$
DDIST	DIST-C4 stream price	$\$/m^3$
DLTISO	LT-ISO stream price	$\$/m^3$
DHVISO	HV-ISO stream price	$\$/m^3$
DLTDST	LT-DIST stream price	$\$/m^3$
DSSFED	SSFEED stream price	$\$/m^3$
DSTAB	STAB feed stream price	$\$/m^3$
DFO	Fuel oil price	$\$/LFEm^3$
F1EFF	C1 reboiler furnace efficiency	—
F2EFF	C2 reboiler furnace efficiency	—
WOBJ	Objective function weight	—

Table C.6: Constraint bounds used in the SPEEDUP real time optimizer input file ICNSTR.DAT.

Variable	Description	Units
A11U	DIST-C4 C5's mole fraction upper bound	lbmol/lbmol
A11L	DIST-C4 C5's mole fraction lower bound	lbmol/lbmol
A31UR	LT-ISO Reid vapour pressure upper bound	psi
A31LT	C2 condenser bubble point temperature lower bound	°F
A41U1	HV-ISO 10% TBP (true boiling point) temperature upper bound	°F
A41L1	HV-ISO 10% TBP temperature lower bound	°F

continued on the next page ...

... Table C.6 continued from the previous page

Variable	Description	Units
A41U9	HV-ISO 90% TBP temperature upper bound	°F
A41L9	HV-ISO 90% TBP temperature lower bound	°F
A51U1	LT-DIST 10% TBP temperature upper bound	°F
A51L1	LT-DIST 10% TBP temperature lower bound	°F
A51U9	LT-DIST 90% TBP temperature upper bound	°F
A51L9	LT-DIST 90% TBP temperature lower bound	°F
F12U	C1 reflux standard liquid volume flow upper bound	bbl/day
F12L	C1 reflux standard liquid volume flow lower bound	bbl/day
F32U	C2 reflux standard liquid volume flow upper bound	bbl/day
F41U	C4 feed stream standard liquid volume flow upper bound	bbl/day
F51U	C5 feed stream standard liquid volume flow upper bound	bbl/day
T11U	C1 tray 3 temperature upper bound	°F
T11L	C1 tray 3 temperature lower bound	°F
T22U	C1 reboiler outlet temperature upper bound	°F
T22L	C1 reboiler outlet temperature lower bound	°F
T31U	C2 tray 2 temperature upper bound	°F
T31L	C2 tray 2 temperature lower bound	°F
T62U	C2 reboiler outlet temperature upper bound	°F
T62L	C2 reboiler outlet temperature lower bound	°F

Table C.7: Optimization free variable ranges used in the SPEEDUP real time optimizer input file IOFREE.DAT.

Variable	Description	Units
C1SPL1.L_OUT%	DIST-C4 stream mole flow	—
C1BT29.L_OUT1%	E1 inlet mole flow	—
C1BT29.L_OUT2%	C1 reboiler mole flow	—
C2COND.T_DROP	C2 condenser subcooled temperature drop	°F
C2FT21.T_IN	E1 outlet temperature	°F
C2BT24.L_OUT1%	SSFEED stream mole flow	—
C2BT24.L_OUT2%	C2 reboiler mole flow	—
C2SPL1.L_OUT1%	C4 feed stream mole flow	—
C2SPL2.L_OUT1%	C5 feed stream mole flow	—
C4REBL.L_OUT%	HV-ISO stream mole flow	—
C5REBL.L_OUT%	LT-DIST stream mole flow	—

C.6.7 RVARNC.DAT

RVARNC.DAT, contains standard deviations used to determine the normal distributions for the random noise to be added to the plant measurements. This input file has the the same format as IVARNC.DAT, with the 28 variances given in the same order as presented in Table C.2.

C.7 Results

Results from the real time optimizer are stored series of directories under the directory results/ for each iteration, e.g., for the first iteration the results are stored in results/itn1/. The results files stored after each iteration are summarised in Table C.8.

Table C.8: SPEEDUP real time optimizer results files.

File	Description
`rtodr6.lis`	Reconciled variables, offsets (reconciled - measured) and data reconciliation error contributions.
`RDSET.DAT`	SPEEDUP simulation settings after data reconciliation
`rtopt6.lis`	Optimization objective function, inequality constraint and set point summary
`ROSET.DAT`	SPEEDUP simulation settings after optimization
`ROSMST.DAT`	ASPEN PLUS simulation settings after optimization
`PLVARS.DAT`	Measures variables from the plant simulation
`IMEAS.DAT`	Measured variables from the plant with noise added for the next iteration of the real time optimizer

C.8 Example

C.8.1 Input Files

The eight input files required to run a SPEEDUP real time optimization run are listed below. They have been printed in two columns to save paper, but appear in a single column in the actual input file.

IMEAS.DAT

```
F01     .20863014E+05
F11     .75209442E+05
F12     .70246257E+04
F13     .88440222E+03
F21     .31955347E+05
F22     .19080626E+05
F31     .15802698E+04
F32     .12862473E+05
F41     .50296623E+04
F42     .38176940E+04
F51     .32645036E+04
F52     .18477370E+04
F61     .29933726E+05
```

```
F62     .12250671E+05
T01     .41295470E+03
T12     .15233838E+03
T13     .96372107E+02
T21     .51601320E+03
T22     .58608488E+03
T23     .47969655E+03
T31     .12835124E+03
T32     .84722528E+02
T41     .24031157E+03
T42     .26107760E+03
T51     .35619866E+03
T52     .42389816E+03
T61     .62344743E+03
T62     .68389991E+03
```

IVARNC.DAT

```
VF01    .10431507E+04
VF11    .37604721E+04
VF12    .35123129E+03
VF13    .44220110E+02
VF21    .15977674E+04
VF22    .95403132E+03
VF31    .79013490E+02
VF32    .64312363E+03
VF41    .25148312E+03
VF42    .19088470E+03
VF51    .16322518E+03
VF52    .92386852E+02
VF61    .14966863E+04
VF62    .61253356E+03
VT01    .20647735E+02
VT12    .76269189E+01
VT13    .48186054E+01
VT21    .25800660E+02
VT22    .29304244E+02
VT23    .23984827E+02
VT31    .64175619E+01
VT32    .42361264E+01
VT41    .12015579E+02
VT42    .13053880E+02
VT51    .17809933E+02
VT52    .21194908E+02
VT61    .31172371E+02
```

```
VT62     .34194995E+02
```

IWGHT.DAT

```
WF01     .12000000E+01
WF11     .12000000E+01
WF12     .12000000E+01
WF13     .12000000E+01
WF21     .12000000E+01
WF22     .12000000E+01
WF31     .12000000E+01
WF32     .12000000E+01
WF41     .12000000E+01
WF42     .12000000E+01
WF51     .12000000E+01
WF52     .12000000E+01
WF61     .12000000E+01
WF62     .12000000E+01
WT01     .12000000E+01
WT12     .12000000E+01
WT13     .12000000E+01
WT21     .12000000E+01
WT22     .12000000E+01
WT23     .12000000E+01
WT31     .12000000E+01
WT32     .12000000E+01
WT41     .12000000E+01
WT42     .12000000E+01
WT51     .12000000E+01
WT52     .12000000E+01
WT61     .12000000E+01
WT62     .12000000E+01
```

IDFREE.DAT

```
     C1COND.Q     .10000000E+01
  C1SPL1.L_OUT    .50000000E+02
   C1FT14.F_IN    .10000000E+03
   C1FT14.T_IN    .20000000E+02
C1BT29.L_OUT1     .10000000E+03
C1BT29.L_OUT2     .10000000E+03
     C2COND.Q     .15000000E+02
C2COND.T_DROP     .15000000E+02
   C2FT21.T_IN    .30000000E+02
```

```
C2BT24.L_OUT1     .10000000E+03
C2BT24.L_OUT2     .10000000E+03
C2SPL1.L_OUT1     .15000000E+03
C2SPL2.L_OUT1     .10000000E+03
 C4REBL.L_OUT     .10000000E+03
 C5REBL.L_OUT     .10000000E+03
```

ISET.DAT

```
MFEED.Z_OUT(1)                          .70368023E-03
MFEED.Z_OUT(2)                          .24683008E-01
MFEED.Z_OUT(3)                          .47364015E-01
MFEED.Z_OUT(4)                          .38724012E-01
MFEED.Z_OUT(5)                          .53531017E-01
MFEED.Z_OUT(6)                          .31279010E-01
MFEED.Z_OUT(7)                          .35193011E-01
MFEED.Z_OUT(8)                          .16149005E-01
MFEED.Z_OUT(9)                          .47256015E-01
MFEED.Z_OUT(10)                         .43562014E-01
MFEED.Z_OUT(11)                         .49649016E-01
MFEED.Z_OUT(12)                         .53071017E-01
MFEED.Z_OUT(13)                         .52326017E-01
MFEED.Z_OUT(14)                         .48963016E-01
MFEED.Z_OUT(15)                         .44418014E-01
MFEED.Z_OUT(16)                         .41240013E-01
MFEED.Z_OUT(17)                         .60769019E-01
MFEED.Z_OUT(18)                         .53782017E-01
MFEED.Z_OUT(19)                         .59831019E-01
MFEED.Z_OUT(20)                         .62687020E-01
MFEED.Z_OUT(21)                         .56050018E-01
MFEED.Z_OUT(22)                         .52921017E-01
MFEED.P_OUT                             .20000000E+03
C1COND.P                                .16300000E+03
C1COND.Q                                .88771527E+01
C1SPL1.L_OUT1                           .11428688E+03
"C1S.TRAY_SS(2)".P                      .16700000E+03
C1FT14.F_IN                             .14871771E+04
C1FT14.T_IN                             .41295470E+03
C1BT29.L_OUT1                           .11995937E+04
C1BT29.L_OUT2                           .20056994E+04
E1.P_OUT                                .34000000E+02
C2COND.P                                .18958100E+02
C2COND.Q                                .38152530E+02
C2COND.T_DROP                           .28033168E+02
"C2S1.TRAY_SS(2)".P                     .22958100E+02
```

C2FT21.T_IN	.47965650E+03
C2BT24.L_OUT1	.52615861E+03
C2BT24.L_OUT2	.13272170E+04
C2SPL1.L_OUT1	.47760940E+03
C2SPL2.L_OUT1	.24821038E+03
C4REBL.L_OUT	.35387055E+03
C5REBL.L_OUT	.13298961E+03
F31.TL_OUT	.84718803E+02

IOBJFN.DAT

PRMRVP	-.18000000E+00
SPCRVP	.97000000E+02
DVAP	.15000000E+00
DDIST	.10007000E+03
DLTISO	.18269000E+03
DHVISO	.19389000E+03
DLTDST	.18052000E+03
DSSFED	.15834000E+03
DSTAB	.15834000E+03
DFO	.82140000E+02
F1EFF	.70000000E+00
F2EFF	.70000000E+00
WOBJ	.12000000E+01

ICNSTR.DAT

A11U	.15000000E+00
A11L	.50000000E-02
A31UR	.14000000E+02
A31LT	.11000000E+03
A41U1	.26600000E+03
A41L1	.17600000E+03
A41U9	.46400000E+03
A41L9	.28400000E+03
A51U1	.55400000E+03
A51L1	.30200000E+03
A51U9	.55400000E+03
A51L9	.30200000E+03
F12U	.90000000E+04
F12L	.10000000E+04
F32U	.25000000E+05
F41U	.10604000E+05
F51U	.60000000E+04

```
T11U     .22000000E+03
T11L     .12500000E+03
T22U     .67500000E+03
T22L     .53000000E+03
T31U     .25000000E+03
T31L     .12000000E+03
T62U     .70000000E+03
T62L     .55000000E+03
```

IOFREE.DAT

```
 C1SPL1.L_OUT%   .50000000E-01
C1BT29.L_OUT1%   .50000000E-01
C1BT29.L_OUT2%   .50000000E-01
C2COND.T_DROP    .50000000E+01
  C2FT21.T_IN    .50000000E+01
C2BT24.L_OUT1%   .50000000E-01
C2BT24.L_OUT2%   .50000000E-01
C2SPL1.L-OUT1%   .50000000E-01
C2SPL2.L_OUT1%   .50000000E-01
 C4REBL.L_OUT%   .50000000E-01
 C5REBL.L_OUT%   .50000000E-01
```

RVARNC.DAT

```
VF01     .10431507E+04
VF11     .37604721E+04
VF12     .35123129E+03
VF13     .44220110E+02
VF21     .15977674E+04
VF22     .95403132E+03
VF31     .79013490E+02
VF32     .64312363E+03
VF41     .25148312E+03
VF42     .19088470E+03
VF51     .16322518E+03
VF52     .92386852E+02
VF61     .14966863E+04
VF62     .61253356E+03
VT01     .20647735E+02
VT12     .76269189E+01
VT13     .48186054E+01
VT21     .25800660E+02
VT22     .29304244E+02
```

```
VT23     .23984827E+02
VT31     .64175619E+01
VT32     .42361264E+01
VT41     .12015579E+02
VT42     .13053880E+02
VT51     .17809933E+02
VT52     .21194908E+02
VT61     .31172371E+02
VT62     .34194995E+02
```

C.8.2 Command Line

If it the SPEEDUP real time optimizer has not already been setup, then type the command:

```
setup
```

When completed, type the command:

```
nohup nice run 1 &
```

to run one cycle of the real time optimizer.

C.8.3 Results

The following results will be present in the `results/itn1/` directory.

rtodr6.lis

```
DATA RECONCILIATION SUMMARY REPORT CREATED BY SPEEDUP ON 09:06:97, 20:33:23
Page 1 of 1
                    ************************************
                    * VARIABLE, OFFSET & ERROR SUMMARY *
                    ************************************

                        VARIABLE        OFFSET         ERROR

       F01 [BBL/DAY]     20457.0926      -405.9214        0.1817
       F11 [CUFT/HR]     63878.2350    -11331.2070       10.8955
       F12 [BBL/DAY]      6638.8516      -385.7741        1.4476
       F13 [BBL/DAY]       771.5942      -112.8080        7.8095
```

F21 [BBL/DAY]	31917.4310	-37.9160	675.7698E-6
F22 [BBL/DAY]	18740.3627	-340.2633	0.1526
F31 [BBL/DAY]	1495.3463	-84.9235	1.3862
F32 [BBL/DAY]	14065.0300	1202.5570	4.1957
F41 [BBL/DAY]	4991.0542	-38.6081	28.2827E-3
F42 [BBL/DAY]	3817.6550	-39.0391E-3	0.0
F51 [BBL/DAY]	3201.8303	-62.6733	0.1769
F52 [BBL/DAY]	1847.5406	-0.1964	5.4246E-6
F61 [BBL/DAY]	29897.7999	-35.9261	691.4245E-6
F62 [BBL/DAY]	11579.8209	-670.8501	1.4394
T01 [F]	426.6606	13.7059	0.5288
T12 [F]	152.3230	-15.3469E-3	4.7035E-6
T13 [F]	91.0478	-5.3243	1.4651
T21 [F]	523.2538	7.2406	94.5091E-3
T22 [F]	586.0582	-26.6524E-3	0.0
T23 [F]	449.6565	-30.0401	1.8824
T31 [F]	151.6020	23.2508	15.7513
T32 [F]	83.0560	-1.6665	0.1857
T41 [F]	275.9938	35.6822	10.5827
T42 [F]	291.5121	30.4345	6.5228
T51 [F]	380.8403	24.6416	2.2972
T52 [F]	434.1996	10.3014	0.2835
T61 [F]	623.4480	618.5440E-6	0.0
T62 [F]	669.6041	-14.2958	0.2097

```
                    ******************************
                    * OBJECTIVE FUNCTION SUMMARY *
                    ******************************

FINAL OBJECTIVE FUNCTION VALUE =      67.5185
```

rtopt6.lis

```
OPTIMIZATION SUMMARY REPORT CREATED BY SPEEDUP ON 10:06:97, 00:55:59
Page 1 of 3
                     ****************************
                     * CONSTRAINT VALUE SUMMARY *
                     ****************************

VARIABLE              LOWER BOUND  VALUE            UPPER BOUND

F12 [BBL/DAY]              5500.0      6634.956996      9000.0
F32 [BBL/DAY]                         12389.577530     25000.0
F41 [BBL/DAY]                          5418.612740     10604.0
F51 [BBL/DAY]                          3503.996662      6000.0
```

T11 [F]	125.0	154.699978	220.0
T22 [F]	500.0	566.744151	675.0
T31 [F]	120.0	167.804627	250.0
T62 [F]	550.0	668.354528	700.0
A11 [PERCENT C5's]	0.5	0.676120	15.0
A31-RVP [PSI]		14.000000	14.0
A31-TBUBBLE [F]	110.0	115.540881	
A41-TBP @ .10 VOL [F]	176.0	210.961598	266.0
A41-TBP @ .90 VOL [F]	284.0	362.141645	464.0
A51-TBP @ .10 VOL [F]	302.0	346.631296	554.0
A51-TBP @ .90 VOL [F]	302.0	498.932721	554.0

****
* OBJECTIVE FUNCTION SUMMARY *
****

FINAL OBJECTIVE FUNCTION VALUE (PROFIT) [$/HR] = 277.363060

TOTAL PRODUCT VALUES [$/HR]	=	21848.829125
TOTAL FEED COSTS [$/HR]	=	21456.451340
TOTAL UTILITY COSTS [$/HR]	=	115.014725

STREAM	STD-VOL FLOWS [BBL/DAY]	COLUMN	HEAT DUTY [MMBTU/HR]
STAB	4785.444846	C1	18.309412
VAP	192.331490	C2	20.530202
DIST-C4	185.050479		
LT-ISO	493.359652		
HV-ISO	884.070411		
LT-DIST	437.753338		
SSFEED	2604.696657		

****
* VARIABLE SUMMARY *
****

VARIABLE	
F01 [BBL/DAY]	20457.093082
F11 [CUFT/HR]	53952.071977
F12 [BBL/DAY]	6634.956996
F13 [BBL/DAY]	791.064361
F21 [BBL/DAY]	31614.428981
F22 [BBL/DAY]	18894.355852
F31 [BBL/DAY]	2109.042031
F32 [BBL/DAY]	12389.577530
F41 [BBL/DAY]	5418.612740
F42 [BBL/DAY]	3779.274709
F51 [BBL/DAY]	3503.996662
F52 [BBL/DAY]	1871.332985
F61 [BBL/DAY]	30341.849577
F62 [BBL/DAY]	11134.706108
T01 [F]	426.660590
T12 [F]	137.200108
T13 [F]	73.811364
T21 [F]	508.585489
T22 [F]	566.744151
T23 [F]	454.656500
T31 [F]	167.804627
T32 [F]	78.056619
T41 [F]	292.933837
T42 [F]	314.595644
T51 [F]	389.997352
T52 [F]	462.118261
T61 [F]	628.022466
T62 [F]	668.354528

OPTIMIZATION SUMMARY REPORT CREATED BY SPEEDUP ON 10:06:97, 00:56:20
Page 3 of 3

****
* SET POINT SUMMARY *
****

C1 REFLUX FLOW (F12) [BBL/DAY]	=	6634.956996
C1 TRAY 3 TEMPERATURE (T11) [F]	=	154.699978
DIST-C4 C5'S MOLE FRACTION (A11) [LBMOL/LBMOL]	=	6.761202E-3
C1 REBOILER TEMPERATURE (T22) [F]	=	566.744151
C2 REFLUX FLOW (F32) [BBL/DAY]	=	12389.577530
C4 FEED FLOW (F41) [BBL/DAY]	=	5418.612740

C5 FEED FLOW (F51) [BBL/DAY]	=	3503.996662
C2 REBOILER TEMPERATURE (T62) [F]	=	668.354528
HV-ISO .10 TBP TEMPERATURE (A41) [F]	=	210.961598
HV-ISO .90 TBP TEMPERATURE (A41) [F]	=	362.141645
LT-DIST .10 TBP TEMPERATURE (A51) [F]	=	346.631296
LT-DIST .90 TBP TEMPERATURE (A51) [F]	=	498.932721

NOTE TO USERS

Page(s) missing in number only; text follows. Microfilmed as received.

294

UMI

Appendix D

ASPEN PLUS Real Time Optimizer Code

D.1 setup

The following executable C-shell script, **setup**, is used to setup the ASPEN PLUS real time optimizer.

```
#!/bin/csh
#
# ASPEN PLUS RTO SETUP SCRIPT
#
# Sets up the Aspen Plus RTO simulation and plant simulations in
# preparation for real time optimzition runs.
#
echo "=== SETTING UP ASPEN PLUS REAL TIME OPTIMIZER SIMULATIONS ==="
cd plant
# Check if input files all exist
if ( !(-e plant6.inp) ) then
  echo "=== SETUP: ERROR: plant/plant6.inp does not exist ==="
  exit
endif
if ( !(-e randdata.f) ) then
  echo "=== SETUP: ERROR: plant/randdata.f does not exist ==="
  exit
endif
echo "=== SETUP: Setting up ASPEN PLUS plant simulation ==="
```

```
aspen plant6 /itonly
echo "=== SETUP: Setting up variable noise FORTRAN ==="
f77 randdata.f -l nag -o randdata.exe
cd ..
cd rto
if ( !(-e rto6.inp) ) then
  echo "=== SETUP: ERROR: rto/rto6.inp does not exist ==="
  exit
endif
echo "=== SETUP: Setting up ASPEN PLUS RTO simulation ==="
aspen rto6 /itonly
echo "=== SETUP: COMPLETED ==="
```

D.2 run

The following executable C-shell script, run, is used to run a given number of cycles of the ASPEN PLUS real time optimizer.

```
#!/bin/csh
#
# ASPEN PLUS RTO RUN SCRIPT
#
# Runs the Aspen Plus RTO simulation and plant simulations.
#
# SYNTAX: run n
#   where n = number of cycles of the real time optimzer to be run
#             (default = 1 if n not given)
#
echo "=== ASPEN PLUS REAL TIME OPTIMIZER SIMULATION ==="
# Check number of cycles to be performed
if ( "$1" >= 1 ) then
  set MAXITN = $1
else
  set MAXITN = 1
endif
set ITN = 1
# Make results directory, deleting old one first
if ( -e results ) then
  rm -r -f results
endif
mkdir results
cd results
mkdir start
```

```
cd ..
while ( $MAXITN >= $ITN )
  echo "=== REAL TIME OPTIMIZER CYCLE $ITN BEGINNING ==="
  cd results
  set RESITN = "itn$ITN"
  mkdir $RESITN
  cd ..
#
# ASPEN RTO6 RUN
#
  cd rto
# Check if input files all exist
  if ( !(-e IMEAS.DAT) ) then
    echo "=== RUN: ERROR: rto/IMEAS.DAT does not exist ==="
    exit
  endif
  if ( !(-e IVARNC.DAT) ) then
    echo "=== RUN: ERROR: rto/IVARNC.DAT does not exist ==="
    exit
  endif
  if ( !(-e IWGHT.DAT) ) then
    echo "=== RUN: ERROR: rto/IWGHT.DAT does not exist ==="
    exit
  endif
  if ( !(-e IDFREE.DAT) ) then
    echo "=== RUN: ERROR: rto/IDFREE.DAT does not exist ==="
    exit
  endif
  if ( !(-e ISMST.DAT) ) then
    echo "=== RUN: ERROR: rto/ISMST.DAT does not exist ==="
    exit
  endif
  if ( !(-e IOBJFN.DAT) ) then
    echo "=== RUN: ERROR: rto/IOBJFN.DAT does not exist ==="
    exit
  endif
  if ( !(-e ICNSTR.DAT) ) then
    echo "=== RUN: ERROR: rto/ICNSTR.DAT does not exist ==="
    exit
  endif
  if ( !(-e IOFREE.DAT) ) then
    echo "=== RUN: ERROR: rto/IOFREE.DAT does not exist ==="
    exit
  endif
# Remove any output files if they exist
```

```
  if ( -e RDVARS.DAT ) then
    rm RDVARS.DAT
  endif
  if ( -e RDOFF.DAT ) then
    rm RDOFF.DAT
  endif
  if ( -e RDERR.DAT ) then
    rm RDERR.DAT
  endif
  if ( -e ROVARS.DAT ) then
    rm ROVARS.DAT
  endif
  if ( -e ROCNST.DAT ) then
    rm ROCNST.DAT
  endif
  if ( -e ROOPFN.DAT ) then
    rm ROOPFN.DAT
  endif
  if ( -e ROSMST.DAT ) then
    rm ROSMST.DAT
  endif
  if ( -e ROSTPT.DAT ) then
    rm ROSTPT.DAT
  endif
# Run Aspen Plus rto6 job
  echo "=== RUN: Starting ASPEN PLUS RTO simulation ==="
  aspen rto6 /sponly
# Check if simulation ran to completion
  if ( !(-e ROSMST.DAT) ) then
    echo "=== RUN: ERROR: an error occured running aspen rto6 ==="
    exit
  endif
# Copy run results to results directory
  cp R*.DAT ../results/$RESITN
  if ("$RESITN" == "itn1") then
    cp IMEAS.DAT ../results/start
    cp ISMST.DAT ../results/start
  endif
# Rename files for next cycle
  cp ROSMST.DAT ISMST.DAT
#
# ASPEN PLANT6 RUN
#
# Set up for plant6 run
  cp ROSMST.DAT ../plant
```

```
  cd ..
  cd plant
# Check if input files all exist
  if ( !(-e ROSMST.DAT) ) then
    echo "=== RUN: ERROR: plant/ROSMST.DAT does not exist ==="
    exit
  endif
# Remove any output files if they exist
  if ( -e PLVARS.DAT ) then
    rm PLVARS.DAT
  endif
# Run Aspen Plus plant6 job
  echo "=== RUN: Starting ASPEN PLUS PLANT simulation ==="
  aspen plant6 /sponly
# Check if simulation ran to completion
  if ( !(-e PLVARS.DAT) ) then
    echo "=== RUN: ERROR: an error occured running aspen plant6 ==="
    exit
  endif
# Copy run results to results directory
  cp PLVARS.DAT ../results/$RESITN
#
# FORTRAN RANDDATA RUN
#
# Check if input files all exist
  if ( !(-e PLVARS.DAT) ) then
    echo "=== RUN: ERROR: plant/PLVARS.DAT does not exist ==="
    exit
  endif
  if ( !(-e RVARNC.DAT) ) then
    echo "=== RUN: ERROR: plant/RVARNC.DAT does not exist ==="
    exit
  endif
# Remove any output files if they exist
  if ( -e IMEAS.DAT ) then
    rm IMEAS.DAT
  endif
# Run FORTRAN randdata.exe job
  echo "=== RUN: Starting randdata.exe ==="
  randdata.exe
# Check if simulation ran to completion
  if ( !(-e IMEAS.DAT) ) then
    echo "=== RUN: ERROR: an error occured running randdata.exe ==="
    exit
  endif
```

```
# Copy run results to results directory
  cp IMEAS.DAT ../results/$RESITN
# Send files to Data Rec run, next cycle
  cp IMEAS.DAT ../rto
  cd ..
  set ITN = `expr $ITN + 1`
end
cp rto/rto6.rep results
cp plant/plant6.rep results
echo "=== $MAXITN REAL TIME OPTIMIZER CYCLES COMPLETED ==="
```

D.3 rto6.inp

The following ASPEN PLUS input code, textttrto6.inp, forms both the data reconciliation and optimization simulations.

```
;
;Input Summary created by ASPEN PLUS Rel. 9.2-1 at 13:31:24 Wed May 14, 1997
;Directory D:\  Filename D:\rto6.inp
;

TITLE 'STABILIZER-SPLITTER REAL TIME OPTIMIZATION'

IN-UNITS ENG VOLUME-FLOW='BBL/DAY' ENTHALPY-FLO='MMBTU/HR'  &
        VOLUME=BBL HEAD=FT HEAT=MMBTU

DEF-STREAMS CONVEN ALL

DIAGNOSTICS
    HISTORY SYS-LEVEL=3 SIM-LEVEL=2 PROP-LEVEL=1 STREAM-LEVEL=1  &
        CONV-LEVEL=4 COST-LEVEL=0 ECON-LEVEL=0
    TERMINAL SIM-LEVEL=2 CONV-LEVEL=4 COST-LEVEL=0 PROP-LEVEL=1  &
        ECON-LEVEL=0 STREAM-LEVEL=1 SYS-LEVEL=3

SIM-OPTIONS
    IN-UNITS ENG
    SIM-OPTIONS FREE-WATER=NO

RUN-CONTROL MAX-TIME=1.0000E+08 MAX-ERRORS=9999 MAX-FORT-ERR=50

DESCRIPTION "
        STABILIZER (C1): MODELLED BY RADFRAC WITH 29 STAGES WITH A PUMPAROUND
          REBOILER.
```

```
        SPLITTER (C2, C4, C5): MODELLED BY PETROFRAC WITH A 24 STAGE
          MAIN COLUMN WITH AN EXTERNAL REBOILER, AND TWO 3 STAGE REBOILED
          SIDESTRIPPERS (C4 AND C5). EXTERNAL REBOILER CONSISTES OF AN FSPLIT
          AND A HEATER.
        PROPERTIES: 15 PSEUDO-COMPONENTS, 8 CONVENTIONAL COMPONENTS,
          PENG-ROBINSON E.O.S. (N.B. WATER IS NOT USED - NO FREE WATER CALCS).
        BASIS: MOLE BASIS USED (LBMOL/HR).
        DATA RECONCILIATION: USING WEIGHTED LEAST SQUARES OBJECTIVE FUNCTION
          SOLVED USING SQP OPTIMIZATION ALGORITHM.
        OPTIMIZATION: MAXIMIZING A PROFIT OBJECTIVE FUNCTION SOLVED USING
          SQP OPTIMIZATION ALGORITHM. "

DATABANKS ASPENPCD  / PURE856  / PURECOMP  / AQUEOUS  /  &
        SOLIDS  / INORGANIC

PROP-SOURCES ASPENPCD  / PURE856  / PURECOMP  / AQUEOUS  /  &
        SOLIDS  / INORGANIC

COMPONENTS
    H2 H2 H2 /
    METHANE CH4 METHANE /
    ETHANE C2H6 ETHANE /
    PROPANE C3H8 PROPANE /
    IBUTANE C4H10-2 IBUTANE /
    BUTANE C4H10-1 BUTANE /
    IPENTANE C5H12-2 IPENTANE /
    PENTANE C5H12-1 PENTANE /
    B45T65 * B45T65 /
    B65T85 * B65T85 /
    B85T105 * B85T105 /
    B105T125 * B105T125 /
    B125T145 * B125T145 /
    B145T165 * B145T165 /
    B165T185 * B165T185 /
    B185T205 * B185T205 /
    B205T235 * B205T235 /
    B235T265 * B235T265 /
    B265T295 * B265T295 /
    B295T325 * B295T325 /
    B325T355 * B325T355 /
    B355T395 * B355T395 /
    B395T510 * B395T510

PC-USER
    PC-DEF API-METH B45T65 NBP=128.49380 GRAV=.64840  &
```

```
        MW=85.98380
    PC-DEF API-METH B65T85 NBP=167.34120 GRAV=.69060  &
        MW=89.15430
    PC-DEF API-METH B85T105 NBP=203.78650 GRAV=.71370  &
        MW=99.91430
    PC-DEF API-METH B105T125 NBP=239.16230 GRAV=.7320  &
        MW=110.25780
    PC-DEF API-METH B125T145 NBP=275.02460 GRAV=.74760  &
        MW=120.44870
    PC-DEF API-METH B145T165 NBP=310.94660 GRAV=.76280  &
        MW=132.08530
    PC-DEF API-METH B165T185 NBP=347.01010 GRAV=.77630  &
        MW=144.87180
    PC-DEF API-METH B185T205 NBP=383.08250 GRAV=.78560  &
        MW=160.93490
    PC-DEF API-METH B205T235 NBP=428.20310 GRAV=.79940  &
        MW=180.13470
    PC-DEF API-METH B235T265 NBP=482.06570 GRAV=.81410  &
        MW=204.11590
    PC-DEF API-METH B265T295 NBP=537.76350 GRAV=.8240  &
        MW=236.50410
    PC-DEF API-METH B295T325 NBP=590.42290 GRAV=.82590  &
        MW=270.94860
    PC-DEF API-METH B325T355 NBP=643.56240 GRAV=.83250  &
        MW=307.94140
    PC-DEF API-METH B355T395 NBP=704.2320 GRAV=.83780  &
        MW=354.40350
    PC-DEF API-METH B395T510 NBP=786.51950 GRAV=.85560  &
        MW=418.92730

FLOWSHEET
    BLOCK C1 IN=STAB OUT=VAP DIST-C4 SPLTFEED C1RFLX C1REBL
    BLOCK C2C4C5 IN=SPLT C2REBL OUT=LT-ISO C2BOT HV-ISO  &
        LT-DIST C2RFLX IC1 IC2
    BLOCK E1 IN=SPLTFEED OUT=SPLT
    BLOCK F2 IN=C2RBFD OUT=C2REBL
    BLOCK C2SPL1 IN=C2BOT OUT=C2RBFD SSFEED

PROPERTIES PENG-ROB SOLU-WATER=2

PROP-SET RVP REIDVP SUBSTREAM=MIXED

PROP-SET STDVOL VLSTDMX UNITS='BBL/DAY' SUBSTREAM=MIXED PHASE=T

PROP-SET TBP-10 TBPT SUBSTREAM=MIXED LVPCT=10.0
```

```
PROP-SET TBP-90 TBPT SUBSTREAM=MIXED LVPCT=90.0

PROP-SET TBUB TBUB SUBSTREAM=MIXED

PROP-SET V-STDVOL VMX UNITS='CUFT/HR' SUBSTREAM=MIXED PHASE=V  &
        TEMP=60.0 PRES=14.6960

PROP-SET VOL VMX UNITS='CUFT/HR' SUBSTREAM=MIXED PHASE=T

STREAM C2REBL
    SUBSTREAM MIXED TEMP=683.899910 PRES=35.65810  &
        MOLE-FLOW=1327.2170
    MOLE-FRAC H2 0.0 / METHANE 1.3091E-18 / ETHANE  &
        3.9177E-15 / PROPANE 3.4897E-11 / IBUTANE 2.3334E-08 /  &
        BUTANE 5.2249E-08 / IPENTANE 1.0680E-06 / PENTANE  &
        6.7323E-07 / B45T65 5.3548E-06 / B65T85 .000013843 /  &
        B85T105 .000042311 / B105T125 .000123760 / B125T145  &
        .000363910 / B145T165 .00117030 / B165T185 .00455110 /  &
        B185T205 .0187060 / B205T235 .1040 / B235T265 .14430  &
        / B265T295 .16840 / B295T325 .17710 / B325T355  &
        .15840 / B355T395 .14950 / B395T510 .073060

STREAM STAB
    SUBSTREAM MIXED TEMP=412.954698 PRES=200.0  &
        MOLE-FLOW=1487.17710
    MOLE-FRAC H2 .000703680 / METHANE .0246830 / ETHANE  &
        .0473640 / PROPANE .0387240 / IBUTANE .0535310 /  &
        BUTANE .0312790 / IPENTANE .0351930 / PENTANE  &
        .0161490 / B45T65 .0472560 / B65T85 .0435620 /  &
        B85T105 .0496490 / B105T125 .0530710 / B125T145  &
        .0523260 / B145T165 .0489630 / B165T185 .0444180 /  &
        B185T205 .041240 / B205T235 .0607690 / B235T265  &
        .0537820 / B265T295 .0598310 / B295T325 .0626870 /  &
        B325T355 .056050 / B355T395 .0529210 / B395T510  &
        .0258480

BLOCK C2SPL1 FSPLIT
    DESCRIPTION "C2 BOTTOMS FLOW SPLITTER"
    MOLE-FLOW C2RBFD 1327.2170

BLOCK E1 HEATER
    IN-UNITS ENG VOLUME-FLOW='BBL/DAY' ENTHALPY-FLO='MMBTU/HR'
    DESCRIPTION "E1 HEAT EXCHANGER"
    PARAM TEMP=479.65650 PRES=34.0
```

```
BLOCK F2 HEATER
    DESCRIPTION "C2 REBOILER FURNACE"
    PARAM TEMP=683.899910 PRES=0.0

BLOCK C1 RADFRAC
    SUBOBJECTS PUMPAROUND = C1REBL
    DESCRIPTION "STABILISER"
    PARAM NSTAGE=29 NPA=1
    PUMPAROUND C1REBL 29 29 NPHASE=2 MOLE-FLOW=2005.69940  &
        TEMP=586.084880
    FEEDS STAB 14 ON-STAGE
    PRODUCTS VAP 1 V / DIST-C4 1 L / SPLTFEED 29 L
    PSEUDO-STREA C1RFLX 1 / C1REBL PA=C1REBL
    P-SPEC 1 163.0 / 2 167.0
    COL-SPECS QN=0.0 DP-COL=6.0 MOLE-RDV=.60262460  &
        MOLE-RR=3.15630830
    T-EST 1 89.5350 / 2 149.040 / 3 169.260 / 4 182.040 /  &
        5 192.140 / 6 200.260 / 7 206.640 / 8 211.760 /  &
        9 216.380 / 10 221.630 / 11 229.350 / 12 243.370 /  &
        13 275.160 / 14 386.670 / 15 389.440 / 16 391.110  &
        / 17 392.160 / 18 392.890 / 19 393.490 / 20  &
        394.060 / 21 394.670 / 22 395.410 / 23 396.360 /  &
        24 397.680 / 25 399.710 / 26 403.380 / 27 411.850  &
        / 28 436.50 / 29 519.830
    L-EST 1 1022.0250 / 2 991.26280 / 3 989.080 / 4  &
        974.47260 / 5 962.86240 / 6 955.53050 / 7 949.85040  &
        / 8 942.2590 / 9 928.30430 / 10 900.91910 / 11  &
        845.60520 / 12 723.95590 / 13 413.96660 / 14  &
        1449.9490 / 15 1500.8610 / 16 1531.4830 / 17  &
        1550.3670 / 18 1563.3520 / 19 1573.6410 / 20  &
        1583.0950 / 21 1593.0030 / 22 1604.5030 / 23  &
        1618.8610 / 24 1637.8240 / 25 1664.3390 / 26  &
        1703.9790 / 27 1764.4650 / 28 1818.9030 / 29  &
        3205.2810
    V-EST 1 173.31230 / 2 1195.3370 / 3 1278.8590 / 4  &
        1276.6760 / 5 1262.0680 / 6 1250.4580 / 7 1243.1260  &
        / 8 1237.4460 / 9 1229.8550 / 10 1215.90 / 11  &
        1188.5150 / 12 1133.2010 / 13 1011.5520 / 14  &
        701.56250 / 15 250.3680 / 16 301.27930 / 17  &
        331.90150 / 18 350.78570 / 19 363.77040 / 20  &
        374.06010 / 21 383.51360 / 22 393.42180 / 23  &
        404.92140 / 24 419.27950 / 25 438.24310 / 26  &
        464.75790 / 27 504.39750 / 28 564.88340 / 29  &
        619.32170
```

```
    TRAY-REPORT TRAY-OPTION=ALL-TRAYS WIDE=YES
    REPORT NOCOMPS NOHYDRAULIC NOENTH NOSPLITS

BLOCK C2C4C5 PETROFRAC
    SUBOBJECTS STRIPPER = C4 C5
    DESCRIPTION "SPLITTER"
    PARAM NPA=0 NSTRIP=2 NSSTRIP=6 NSTAGE=24 HYDRAULIC=NO
    FEEDS SPLT 21 ON-STAGE / C2REBL 24 ON-STAGE
    PRODUCTS LT-ISO 1 L / C2BOT 24 L
    COL-SPECS CONDENSER=SUBCOOLED REBOILER=NONE-BOTFEED  &
        T1=84.7225280 MOLE-D=186.57490 DP-COL=12.70
    P-SPEC 1 18.95810 / 2 22.95810
    T-EST 1 84.7230 / 2 150.930 / 3 169.420 / 4 181.130 /  &
        5 188.750 / 6 194.010 / 7 197.990 / 8 201.440 /  &
        9 204.930 / 10 208.970 / 11 214.20 / 12 221.490 /  &
        13 232.330 / 14 249.640 / 15 278.020 / 16 320.730  &
        / 17 358.980 / 18 388.210 / 19 415.850 / 20  &
        441.860 / 21 483.390 / 22 531.190 / 23 570.180 /  &
        24 626.450
    V-EST 1 0.0 / 2 2776.1290 / 3 3094.8350 / 4 3095.9990  &
        / 5 3101.710 / 6 3106.5220 / 7 3108.8040 / 8  &
        3107.4250 / 9 3100.7670 / 10 3086.2130 / 11  &
        3059.6610 / 12 3014.5630 / 13 2939.0650 / 14  &
        2809.8090 / 15 2473.5240 / 16 2206.0180 / 17  &
        2037.2320 / 18 1833.3290 / 19 1747.6360 / 20  &
        1647.220 / 21 1439.3880 / 22 772.39920 / 23  &
        965.84870 / 24 953.74740
    L-EST 1 2776.1290 / 2 2908.260 / 3 2909.4240 / 4  &
        2915.1350 / 5 2919.9470 / 6 2922.2290 / 7 2920.850  &
        / 8 2914.1920 / 9 2899.6380 / 10 2873.0860 / 11  &
        2827.9890 / 12 2752.490 / 13 2623.2340 / 14  &
        2410.6880 / 15 2143.1810 / 16 1496.7840 / 17  &
        1408.0990 / 18 1322.4060 / 19 973.77810 / 20  &
        765.94590 / 21 1298.5380 / 22 1491.9880 / 23  &
        1479.8870 / 24 1853.3340
    PSEUDO-STREA C2RFLX STAGE=1
    STRIPPER C4 NSTAGE=3 LDRAW=15 VRETURN=14 PRODUCT=HV-ISO  &
        MOLE-DRAW=477.61110 Q-REB=1.90090
    STRIPPER C5 NSTAGE=3 LDRAW=18 VRETURN=17 PRODUCT=LT-DIST  &
        MOLE-DRAW=248.21130 Q-REB=2.39920
    STR-T-EST C4 1 278.790 / C4 2 282.20 / C4 3 292.030 /  &
        C5 1 402.780 / C5 2 414.880 / C5 3 429.50
    STR-V-EST C4 1 123.70 / C4 2 122.70 / C4 3 123.40 /  &
        C5 1 115.20 / C5 2 125.60 / C5 3 130.40
    STR-L-EST C4 1 476.60 / C4 2 477.20 / C4 3 353.90 /  &
```

```
        C5 1 258.60 / C5 2 263.40 / C5 3 133.0
    STR-PSEUDO-S C4 IC1 SOURCE=FEED DRAW=15 / C5 IC2  &
        SOURCE=FEED DRAW=18
    TRAY-REPORT TRAY-OPTION=ALL-TRAYS WIDE=YES
    REPORT RESULTS NOCOMPS NOHYDRAULIC NOENTH

FORTRAN DRDATA
F     COMMON / USRSF1 / SF01,SF11,SF12,SF13,SF21,SF22
F     COMMON / USRSF2 / SF31,SF32,SF41,SF42,SF51,SF52,SF61,SF62
F     COMMON / USRST1 / ST01,ST12,ST13,ST21,ST22,ST23
F     COMMON / USRST2 / ST31,ST32,ST41,ST42,ST51,ST52,ST61,ST62
F     COMMON / USREF1 / EF01,EF11,EF12,EF13,EF21,EF22
F     COMMON / USREF2 / EF31,EF32,EF41,EF42,EF51,EF52,EF61,EF62
F     COMMON / USRET1 / ET01,ET12,ET13,ET21,ET22,ET23
F     COMMON / USRET2 / ET31,ET32,ET41,ET42,ET51,ET52,ET61,ET62
F     LOGICAL THERE
    DEFINE F01 STREAM-VAR STREAM=STAB SUBSTREAM=MIXED  &
        VARIABLE=STDVOL-FLOW
    DEFINE F11 STREAM-PROP STREAM=VAP PROPERTY=V-STDVOL
    DEFINE F12 STREAM-VAR STREAM=C1RFLX SUBSTREAM=MIXED  &
        VARIABLE=STDVOL-FLOW
    DEFINE F13 STREAM-VAR STREAM=DIST-C4 SUBSTREAM=MIXED  &
        VARIABLE=STDVOL-FLOW
    DEFINE F21 STREAM-VAR STREAM=C1REBL SUBSTREAM=MIXED  &
        VARIABLE=STDVOL-FLOW
    DEFINE F22 STREAM-VAR STREAM=SPLTFEED SUBSTREAM=MIXED  &
        VARIABLE=STDVOL-FLOW
    DEFINE F31 STREAM-VAR STREAM=LT-ISO SUBSTREAM=MIXED  &
        VARIABLE=STDVOL-FLOW
    DEFINE F32 STREAM-VAR STREAM=C2RFLX SUBSTREAM=MIXED  &
        VARIABLE=STDVOL-FLOW
    DEFINE F41 STREAM-VAR STREAM=IC1 SUBSTREAM=MIXED  &
        VARIABLE=STDVOL-FLOW
    DEFINE F42 STREAM-VAR STREAM=HV-ISO SUBSTREAM=MIXED  &
        VARIABLE=STDVOL-FLOW
    DEFINE F51 STREAM-VAR STREAM=IC2 SUBSTREAM=MIXED  &
        VARIABLE=STDVOL-FLOW
    DEFINE F52 STREAM-VAR STREAM=LT-DIST SUBSTREAM=MIXED  &
        VARIABLE=STDVOL-FLOW
    DEFINE F61 STREAM-VAR STREAM=C2REBL SUBSTREAM=MIXED  &
        VARIABLE=STDVOL-FLOW
    DEFINE F62 STREAM-VAR STREAM=SSFEED SUBSTREAM=MIXED  &
        VARIABLE=STDVOL-FLOW
    DEFINE T01 STREAM-VAR STREAM=STAB SUBSTREAM=MIXED  &
        VARIABLE=TEMP
```

```
      DEFINE T12 BLOCK-VAR BLOCK=C1 VARIABLE=TEMP  &
          SENTENCE=PROFILE ID1=2
      DEFINE T13 STREAM-VAR STREAM=DIST-C4 SUBSTREAM=MIXED  &
          VARIABLE=TEMP
      DEFINE T21 STREAM-VAR STREAM=SPLTFEED SUBSTREAM=MIXED  &
          VARIABLE=TEMP
      DEFINE T22 STREAM-VAR STREAM=C1REBL SUBSTREAM=MIXED  &
          VARIABLE=TEMP
      DEFINE T23 STREAM-VAR STREAM=SPLT SUBSTREAM=MIXED  &
          VARIABLE=TEMP
      DEFINE T31 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=TEMP  &
          SENTENCE=PROFILE ID1=2
      DEFINE T32 STREAM-VAR STREAM=LT-ISO SUBSTREAM=MIXED  &
          VARIABLE=TEMP
      DEFINE T41 STREAM-VAR STREAM=IC1 SUBSTREAM=MIXED  &
          VARIABLE=TEMP
      DEFINE T42 STREAM-VAR STREAM=HV-ISO SUBSTREAM=MIXED  &
          VARIABLE=TEMP
      DEFINE T51 STREAM-VAR STREAM=IC2 SUBSTREAM=MIXED  &
          VARIABLE=TEMP
      DEFINE T52 STREAM-VAR STREAM=LT-DIST SUBSTREAM=MIXED  &
          VARIABLE=TEMP
      DEFINE T61 STREAM-VAR STREAM=SSFEED SUBSTREAM=MIXED  &
          VARIABLE=TEMP
      DEFINE T62 STREAM-VAR STREAM=C2REBL SUBSTREAM=MIXED  &
          VARIABLE=TEMP
F       WRITE(NTERM,*) '+++ DATA REC RESULTS FORTRAN STARTED +++'
C WRITE TO FILE MEASURED VARIABLES VALUES AFTER DATA REC
F       INQUIRE (FILE='RDVARS.DAT', EXIST=THERE)
F       IF (THERE) THEN
F         WRITE(NTERM,*) '*** ERROR: RDVARS.DAT ALREADY EXISTS ***'
F         WRITE(NRPT,*) '*** ERROR: RDVARS.DAT ALREADY EXISTED ***'
F         WRITE(NHSTRY,*) '*** ERROR: RDVARS.DAT ALREADY EXISTS ***'
F         GO TO 50
F       ENDIF
F       OPEN (UNIT=63, STATUS='NEW', FILE='RDVARS.DAT')
F       WRITE(63,100) 'F01', F01
F       WRITE(63,100) 'F11', F11
F       WRITE(63,100) 'F12', F12
F       WRITE(63,100) 'F13', F13
F       WRITE(63,100) 'F21', F21
F       WRITE(63,100) 'F22', F22
F       WRITE(63,100) 'F31', F31
F       WRITE(63,100) 'F32', F32
F       WRITE(63,100) 'F41', F41
```

```
F     WRITE(63,100) 'F42', F42
F     WRITE(63,100) 'F51', F51
F     WRITE(63,100) 'F52', F52
F     WRITE(63,100) 'F61', F61
F     WRITE(63,100) 'F62', F62
F     WRITE(63,100) 'T01', T01
F     WRITE(63,100) 'T12', T12
F     WRITE(63,100) 'T13', T13
F     WRITE(63,100) 'T21', T21
F     WRITE(63,100) 'T22', T22
F     WRITE(63,100) 'T23', T23
F     WRITE(63,100) 'T31', T31
F     WRITE(63,100) 'T32', T32
F     WRITE(63,100) 'T41', T41
F     WRITE(63,100) 'T42', T42
F     WRITE(63,100) 'T51', T51
F     WRITE(63,100) 'T52', T52
F     WRITE(63,100) 'T61', T61
F     WRITE(63,100) 'T62', T62
F     CLOSE (UNIT=63)
F     WRITE(NTERM,*) 'RDVARS.DAT WRITTEN SUCCESSFULLY'
C WRITE TO FILE MEASURED TO RECONCILED VAR OFFSETS AFTER DATA REC
F  50 INQUIRE (FILE='RDOFF.DAT', EXIST=THERE)
F     IF (THERE) THEN
F       WRITE(NTERM,*) '*** ERROR: RDOFF.DAT ALREADY EXISTS ***'
F       WRITE(NRPT,*) '*** ERROR: RDOFF.DAT ALREADY EXISTED ***'
F       WRITE(NHSTRY,*) '*** ERROR: RDOFF.DAT ALREADY EXISTS ***'
F       GO TO 52
F     ENDIF
F     OPEN (UNIT=63, STATUS='NEW', FILE='RDOFF.DAT')
F     OFFF01 = F01-SF01
F     OFFF11 = F11-SF11
F     OFFF12 = F12-SF12
F     OFFF13 = F13-SF13
F     OFFF21 = F21-SF21
F     OFFF22 = F22-SF22
F     OFFF31 = F31-SF31
F     OFFF32 = F32-SF32
F     OFFF41 = F41-SF41
F     OFFF42 = F42-SF42
F     OFFF51 = F51-SF51
F     OFFF52 = F52-SF52
F     OFFF61 = F61-SF61
F     OFFF62 = F62-SF62
F     OFFT01 = T01-ST01
```

```
F     OFFT12 = T12-ST12
F     OFFT13 = T13-ST13
F     OFFT21 = T21-ST21
F     OFFT22 = T22-ST22
F     OFFT23 = T23-ST23
F     OFFT31 = T31-ST31
F     OFFT32 = T32-ST32
F     OFFT41 = T41-ST41
F     OFFT42 = T42-ST42
F     OFFT51 = T51-ST51
F     OFFT52 = T52-ST52
F     OFFT61 = T61-ST61
F     OFFT62 = T62-ST62
F     WRITE(63,100) 'OFFF01', OFFF01
F     WRITE(63,100) 'OFFF11', OFFF11
F     WRITE(63,100) 'OFFF12', OFFF12
F     WRITE(63,100) 'OFFF13', OFFF13
F     WRITE(63,100) 'OFFF21', OFFF21
F     WRITE(63,100) 'OFFF22', OFFF22
F     WRITE(63,100) 'OFFF31', OFFF31
F     WRITE(63,100) 'OFFF32', OFFF32
F     WRITE(63,100) 'OFFF41', OFFF41
F     WRITE(63,100) 'OFFF42', OFFF42
F     WRITE(63,100) 'OFFF51', OFFF51
F     WRITE(63,100) 'OFFF52', OFFF52
F     WRITE(63,100) 'OFFF61', OFFF61
F     WRITE(63,100) 'OFFF62', OFFF62
F     WRITE(63,100) 'OFFT01', OFFT01
F     WRITE(63,100) 'OFFT12', OFFT12
F     WRITE(63,100) 'OFFT13', OFFT13
F     WRITE(63,100) 'OFFT21', OFFT21
F     WRITE(63,100) 'OFFT22', OFFT22
F     WRITE(63,100) 'OFFT23', OFFT23
F     WRITE(63,100) 'OFFT31', OFFT31
F     WRITE(63,100) 'OFFT32', OFFT32
F     WRITE(63,100) 'OFFT41', OFFT41
F     WRITE(63,100) 'OFFT42', OFFT42
F     WRITE(63,100) 'OFFT51', OFFT51
F     WRITE(63,100) 'OFFT52', OFFT52
F     WRITE(63,100) 'OFFT61', OFFT61
F     WRITE(63,100) 'OFFT62', OFFT62
F     CLOSE (UNIT=63)
F     WRITE(NTERM,*) 'RDOFF.DAT WRITTEN SUCCESSFULLY'
C WRITE TO FILE DATA RECONCILIATION ERROR TERMS
F  52 INQUIRE (FILE='RDERR.DAT', EXIST=THERE)
```

```
F     IF (THERE) THEN
F       WRITE(NTERM,*) '*** ERROR: RDERR.DAT ALREADY EXISTS ***'
F       WRITE(NRPT,*) '*** ERROR: RDERR.DAT ALREADY EXISTED ***'
F       WRITE(NHSTRY,*) '*** ERROR: RDERR.DAT ALREADY EXISTS ***'
F       GO TO 54
F     ENDIF
F     OPEN (UNIT=63, STATUS='NEW', FILE='RDERR.DAT')
F     WRITE(63,100) 'ERRF01', EF01
F     WRITE(63,100) 'ERRF11', EF11
F     WRITE(63,100) 'ERRF12', EF12
F     WRITE(63,100) 'ERRF13', EF13
F     WRITE(63,100) 'ERRF21', EF21
F     WRITE(63,100) 'ERRF22', EF22
F     WRITE(63,100) 'ERRF31', EF31
F     WRITE(63,100) 'ERRF32', EF32
F     WRITE(63,100) 'ERRF41', EF41
F     WRITE(63,100) 'ERRF42', EF42
F     WRITE(63,100) 'ERRF51', EF51
F     WRITE(63,100) 'ERRF52', EF52
F     WRITE(63,100) 'ERRF61', EF61
F     WRITE(63,100) 'ERRF62', EF62
F     WRITE(63,100) 'ERRT01', ET01
F     WRITE(63,100) 'ERRT12', ET12
F     WRITE(63,100) 'ERRT13', ET13
F     WRITE(63,100) 'ERRT21', ET21
F     WRITE(63,100) 'ERRT22', ET22
F     WRITE(63,100) 'ERRT23', ET23
F     WRITE(63,100) 'ERRT31', ET31
F     WRITE(63,100) 'ERRT32', ET32
F     WRITE(63,100) 'ERRT41', ET41
F     WRITE(63,100) 'ERRT42', ET42
F     WRITE(63,100) 'ERRT51', ET51
F     WRITE(63,100) 'ERRT52', ET52
F     WRITE(63,100) 'ERRT61', ET61
F     WRITE(63,100) 'ERRT62', ET62
F     CLOSE (UNIT=63)
F     WRITE(NTERM,*) 'RDERR.DAT WRITTEN SUCCESSFULLY'
F  54 WRITE(NTERM,*) '+++ DATA REC RESULTS FORTRAN COMPLETED +++'
C FORMAT STATEMENTS
F 100 FORMAT(A13, 2X, E15.8)
    EXECUTE LAST

FORTRAN DRSPCSET
F     COMMON / USRSF1 / SF01,SF11,SF12,SF13,SF21,SF22
F     COMMON / USRSF2 / SF31,SF32,SF41,SF42,SF51,SF52,SF61,SF62
```

```
F     COMMON / USRST1 / ST01,ST12,ST13,ST21,ST22,ST23
F     COMMON / USRST2 / ST31,ST32,ST41,ST42,ST51,ST52,ST61,ST62
F     COMMON / USRVF1 / VF01,VF11,VF12,VF13,VF21,VF22
F     COMMON / USRVF2 / VF31,VF32,VF41,VF42,VF51,VF52,VF61,VF62
F     COMMON / USRVT1 / VT01,VT12,VT13,VT21,VT22,VT23
F     COMMON / USRVT2 / VT31,VT32,VT41,VT42,VT51,VT52,VT61,VT62
F     COMMON / USRWF1 / WF01,WF11,WF12,WF13,WF21,WF22
F     COMMON / USRWF2 / WF31,WF32,WF41,WF42,WF51,WF52,WF61,WF62
F     COMMON / USRWT1 / WT01,WT12,WT13,WT21,WT22,WT23
F     COMMON / USRWT2 / WT31,WT32,WT41,WT42,WT51,WT52,WT61,WT62
F     COMMON / USRB1 / BSTABF,BSTABT,BC1RR,BC1RDV,BC1RBF,BC1RBT
F     COMMON / USRB2 / BE1T,BC2MD,BC2RBF,BC2RBT,BC2SCT,BC4QN
F     COMMON / USRB3 / BC5QN,BC2C4F,BC2C5F
F     LOGICAL THERE
      DEFINE STABF STREAM-VAR STREAM=STAB SUBSTREAM=MIXED  &
          VARIABLE=MOLE-FLOW
      DEFINE STABT STREAM-VAR STREAM=STAB SUBSTREAM=MIXED  &
          VARIABLE=TEMP
      DEFINE STABP STREAM-VAR STREAM=STAB SUBSTREAM=MIXED  &
          VARIABLE=PRES
      DEFINE STBZ1 MOLE-FRAC STREAM=STAB SUBSTREAM=MIXED  &
          COMPONENT=H2
      DEFINE STBZ2 MOLE-FRAC STREAM=STAB SUBSTREAM=MIXED  &
          COMPONENT=METHANE
      DEFINE STBZ3 MOLE-FRAC STREAM=STAB SUBSTREAM=MIXED  &
          COMPONENT=ETHANE
      DEFINE STBZ4 MOLE-FRAC STREAM=STAB SUBSTREAM=MIXED  &
          COMPONENT=PROPANE
      DEFINE STBZ5 MOLE-FRAC STREAM=STAB SUBSTREAM=MIXED  &
          COMPONENT=IBUTANE
      DEFINE STBZ6 MOLE-FRAC STREAM=STAB SUBSTREAM=MIXED  &
          COMPONENT=BUTANE
      DEFINE STBZ7 MOLE-FRAC STREAM=STAB SUBSTREAM=MIXED  &
          COMPONENT=IPENTANE
      DEFINE STBZ8 MOLE-FRAC STREAM=STAB SUBSTREAM=MIXED  &
          COMPONENT=PENTANE
      DEFINE STBZ9 MOLE-FRAC STREAM=STAB SUBSTREAM=MIXED  &
          COMPONENT=B45T65
      DEFINE STBZ10 MOLE-FRAC STREAM=STAB SUBSTREAM=MIXED  &
          COMPONENT=B65T85
      DEFINE STBZ11 MOLE-FRAC STREAM=STAB SUBSTREAM=MIXED  &
          COMPONENT=B85T105
      DEFINE STBZ12 MOLE-FRAC STREAM=STAB SUBSTREAM=MIXED  &
          COMPONENT=B105T125
      DEFINE STBZ13 MOLE-FRAC STREAM=STAB SUBSTREAM=MIXED  &
```

```
    COMPONENT=B125T145
DEFINE STBZ14 MOLE-FRAC STREAM=STAB SUBSTREAM=MIXED  &
    COMPONENT=B145T165
DEFINE STBZ15 MOLE-FRAC STREAM=STAB SUBSTREAM=MIXED  &
    COMPONENT=B165T185
DEFINE STBZ16 MOLE-FRAC STREAM=STAB SUBSTREAM=MIXED  &
    COMPONENT=B185T205
DEFINE STBZ17 MOLE-FRAC STREAM=STAB SUBSTREAM=MIXED  &
    COMPONENT=B205T235
DEFINE STBZ18 MOLE-FRAC STREAM=STAB SUBSTREAM=MIXED  &
    COMPONENT=B235T265
DEFINE STBZ19 MOLE-FRAC STREAM=STAB SUBSTREAM=MIXED  &
    COMPONENT=B265T295
DEFINE STBZ20 MOLE-FRAC STREAM=STAB SUBSTREAM=MIXED  &
    COMPONENT=B295T325
DEFINE STBZ21 MOLE-FRAC STREAM=STAB SUBSTREAM=MIXED  &
    COMPONENT=B325T355
DEFINE STBZ22 MOLE-FRAC STREAM=STAB SUBSTREAM=MIXED  &
    COMPONENT=B355T395
DEFINE STBZ23 MOLE-FRAC STREAM=STAB SUBSTREAM=MIXED  &
    COMPONENT=B395T510
DEFINE C2RBF1 BLOCK-VAR BLOCK=C2SPL1 SENTENCE=MOLE-FLOW  &
    VARIABLE=FLOW ID1=C2RBFD
DEFINE C2RBF2 STREAM-VAR STREAM=C2REBL SUBSTREAM=MIXED  &
    VARIABLE=MOLE-FLOW
DEFINE C2RBT1 BLOCK-VAR BLOCK=F2 VARIABLE=TEMP  &
    SENTENCE=PARAM
DEFINE C2RBT2 STREAM-VAR STREAM=C2REBL SUBSTREAM=MIXED  &
    VARIABLE=TEMP
DEFINE C2RBP1 BLOCK-VAR BLOCK=F2 VARIABLE=PRES  &
    SENTENCE=PARAM
DEFINE C2RBP2 STREAM-VAR STREAM=C2REBL SUBSTREAM=MIXED  &
    VARIABLE=PRES
DEFINE C2RZ1 MOLE-FRAC STREAM=C2REBL SUBSTREAM=MIXED  &
    COMPONENT=H2
DEFINE C2RZ2 MOLE-FRAC STREAM=C2REBL SUBSTREAM=MIXED  &
    COMPONENT=METHANE
DEFINE C2RZ3 MOLE-FRAC STREAM=C2REBL SUBSTREAM=MIXED  &
    COMPONENT=ETHANE
DEFINE C2RZ4 MOLE-FRAC STREAM=C2REBL SUBSTREAM=MIXED  &
    COMPONENT=PROPANE
DEFINE C2RZ5 MOLE-FRAC STREAM=C2REBL SUBSTREAM=MIXED  &
    COMPONENT=IBUTANE
DEFINE C2RZ6 MOLE-FRAC STREAM=C2REBL SUBSTREAM=MIXED  &
    COMPONENT=BUTANE
```

```
DEFINE C2RZ7 MOLE-FRAC STREAM=C2REBL SUBSTREAM=MIXED  &
    COMPONENT=IPENTANE
DEFINE C2RZ8 MOLE-FRAC STREAM=C2REBL SUBSTREAM=MIXED  &
    COMPONENT=PENTANE
DEFINE C2RZ9 MOLE-FRAC STREAM=C2REBL SUBSTREAM=MIXED  &
    COMPONENT=B45T65
DEFINE C2RZ10 MOLE-FRAC STREAM=C2REBL SUBSTREAM=MIXED  &
    COMPONENT=B65T85
DEFINE C2RZ11 MOLE-FRAC STREAM=C2REBL SUBSTREAM=MIXED  &
    COMPONENT=B85T105
DEFINE C2RZ12 MOLE-FRAC STREAM=C2REBL SUBSTREAM=MIXED  &
    COMPONENT=B105T125
DEFINE C2RZ13 MOLE-FRAC STREAM=C2REBL SUBSTREAM=MIXED  &
    COMPONENT=B125T145
DEFINE C2RZ14 MOLE-FRAC STREAM=C2REBL SUBSTREAM=MIXED  &
    COMPONENT=B145T165
DEFINE C2RZ15 MOLE-FRAC STREAM=C2REBL SUBSTREAM=MIXED  &
    COMPONENT=B165T185
DEFINE C2RZ16 MOLE-FRAC STREAM=C2REBL SUBSTREAM=MIXED  &
    COMPONENT=B185T205
DEFINE C2RZ17 MOLE-FRAC STREAM=C2REBL SUBSTREAM=MIXED  &
    COMPONENT=B205T235
DEFINE C2RZ18 MOLE-FRAC STREAM=C2REBL SUBSTREAM=MIXED  &
    COMPONENT=B235T265
DEFINE C2RZ19 MOLE-FRAC STREAM=C2REBL SUBSTREAM=MIXED  &
    COMPONENT=B265T295
DEFINE C2RZ20 MOLE-FRAC STREAM=C2REBL SUBSTREAM=MIXED  &
    COMPONENT=B295T325
DEFINE C2RZ21 MOLE-FRAC STREAM=C2REBL SUBSTREAM=MIXED  &
    COMPONENT=B325T355
DEFINE C2RZ22 MOLE-FRAC STREAM=C2REBL SUBSTREAM=MIXED  &
    COMPONENT=B355T395
DEFINE C2RZ23 MOLE-FRAC STREAM=C2REBL SUBSTREAM=MIXED  &
    COMPONENT=B395T510
DEFINE E1T BLOCK-VAR BLOCK=E1 VARIABLE=TEMP SENTENCE=PARAM
DEFINE E1P BLOCK-VAR BLOCK=E1 VARIABLE=PRES SENTENCE=PARAM
DEFINE C1RR BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-RR  &
    SENTENCE=COL-SPECS
DEFINE C1RDV BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-RDV  &
    SENTENCE=COL-SPECS
DEFINE C1RBFL BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=PUMPAROUND ID1=C1REBL
DEFINE C1RBT BLOCK-VAR BLOCK=C1 VARIABLE=TEMP  &
    SENTENCE=PUMPAROUND ID1=C1REBL
DEFINE C1P1 BLOCK-VAR BLOCK=C1 VARIABLE=PRES  &
```

```
    SENTENCE=P-SPEC ID1=1
DEFINE C1P2 BLOCK-VAR BLOCK=C1 VARIABLE=PRES  &
    SENTENCE=P-SPEC ID1=2
DEFINE C1DP BLOCK-VAR BLOCK=C1 VARIABLE=DP-COL  &
    SENTENCE=COL-SPECS
DEFINE C2MD BLOCK-VAR BLOCK=C2C4C5 VARIABLE=MOLE-D  &
    SENTENCE=COL-SPECS
DEFINE C2SC BLOCK-VAR BLOCK=C2C4C5 VARIABLE=T1  &
    SENTENCE=COL-SPECS
DEFINE C2P1 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=PRES  &
    SENTENCE=P-SPEC ID1=1
DEFINE C2P2 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=PRES  &
    SENTENCE=P-SPEC ID1=2
DEFINE C2DP BLOCK-VAR BLOCK=C2C4C5 VARIABLE=DP-COL  &
    SENTENCE=COL-SPECS
DEFINE C2IC1 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=MOLE-DRAW  &
    SENTENCE=STRIPPER ID1=C4
DEFINE C2IC2 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=MOLE-DRAW  &
    SENTENCE=STRIPPER ID1=C5
DEFINE C4QN BLOCK-VAR BLOCK=C2C4C5 VARIABLE=Q-REB  &
    SENTENCE=STRIPPER ID1=C4
DEFINE C5QN BLOCK-VAR BLOCK=C2C4C5 VARIABLE=Q-REB  &
    SENTENCE=STRIPPER ID1=C5
DEFINE C1T1 BLOCK-VAR BLOCK=C1 VARIABLE=TEMP SENTENCE=T-EST  &
    ID1=1
DEFINE C1T2 BLOCK-VAR BLOCK=C1 VARIABLE=TEMP SENTENCE=T-EST  &
    ID1=2
DEFINE C1T3 BLOCK-VAR BLOCK=C1 VARIABLE=TEMP SENTENCE=T-EST  &
    ID1=3
DEFINE C1T4 BLOCK-VAR BLOCK=C1 VARIABLE=TEMP SENTENCE=T-EST  &
    ID1=4
DEFINE C1T5 BLOCK-VAR BLOCK=C1 VARIABLE=TEMP SENTENCE=T-EST  &
    ID1=5
DEFINE C1T6 BLOCK-VAR BLOCK=C1 VARIABLE=TEMP SENTENCE=T-EST  &
    ID1=6
DEFINE C1T7 BLOCK-VAR BLOCK=C1 VARIABLE=TEMP SENTENCE=T-EST  &
    ID1=7
DEFINE C1T8 BLOCK-VAR BLOCK=C1 VARIABLE=TEMP SENTENCE=T-EST  &
    ID1=8
DEFINE C1T9 BLOCK-VAR BLOCK=C1 VARIABLE=TEMP SENTENCE=T-EST  &
    ID1=9
DEFINE C1T10 BLOCK-VAR BLOCK=C1 VARIABLE=TEMP  &
    SENTENCE=T-EST ID1=10
DEFINE C1T11 BLOCK-VAR BLOCK=C1 VARIABLE=TEMP  &
    SENTENCE=T-EST ID1=11
```

```
DEFINE C1T12 BLOCK-VAR BLOCK=C1 VARIABLE=TEMP  &
    SENTENCE=T-EST ID1=12
DEFINE C1T13 BLOCK-VAR BLOCK=C1 VARIABLE=TEMP  &
    SENTENCE=T-EST ID1=13
DEFINE C1T14 BLOCK-VAR BLOCK=C1 VARIABLE=TEMP  &
    SENTENCE=T-EST ID1=14
DEFINE C1T15 BLOCK-VAR BLOCK=C1 VARIABLE=TEMP  &
    SENTENCE=T-EST ID1=15
DEFINE C1T16 BLOCK-VAR BLOCK=C1 VARIABLE=TEMP  &
    SENTENCE=T-EST ID1=16
DEFINE C1T17 BLOCK-VAR BLOCK=C1 VARIABLE=TEMP  &
    SENTENCE=T-EST ID1=17
DEFINE C1T18 BLOCK-VAR BLOCK=C1 VARIABLE=TEMP  &
    SENTENCE=T-EST ID1=18
DEFINE C1T19 BLOCK-VAR BLOCK=C1 VARIABLE=TEMP  &
    SENTENCE=T-EST ID1=19
DEFINE C1T20 BLOCK-VAR BLOCK=C1 VARIABLE=TEMP  &
    SENTENCE=T-EST ID1=20
DEFINE C1T21 BLOCK-VAR BLOCK=C1 VARIABLE=TEMP  &
    SENTENCE=T-EST ID1=21
DEFINE C1T22 BLOCK-VAR BLOCK=C1 VARIABLE=TEMP  &
    SENTENCE=T-EST ID1=22
DEFINE C1T23 BLOCK-VAR BLOCK=C1 VARIABLE=TEMP  &
    SENTENCE=T-EST ID1=23
DEFINE C1T24 BLOCK-VAR BLOCK=C1 VARIABLE=TEMP  &
    SENTENCE=T-EST ID1=24
DEFINE C1T25 BLOCK-VAR BLOCK=C1 VARIABLE=TEMP  &
    SENTENCE=T-EST ID1=25
DEFINE C1T26 BLOCK-VAR BLOCK=C1 VARIABLE=TEMP  &
    SENTENCE=T-EST ID1=26
DEFINE C1T27 BLOCK-VAR BLOCK=C1 VARIABLE=TEMP  &
    SENTENCE=T-EST ID1=27
DEFINE C1T28 BLOCK-VAR BLOCK=C1 VARIABLE=TEMP  &
    SENTENCE=T-EST ID1=28
DEFINE C1T29 BLOCK-VAR BLOCK=C1 VARIABLE=TEMP  &
    SENTENCE=T-EST ID1=29
DEFINE C2T1 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=TEMP  &
    SENTENCE=T-EST ID1=1
DEFINE C2T2 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=TEMP  &
    SENTENCE=T-EST ID1=2
DEFINE C2T3 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=TEMP  &
    SENTENCE=T-EST ID1=3
DEFINE C2T4 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=TEMP  &
    SENTENCE=T-EST ID1=4
DEFINE C2T5 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=TEMP  &
```

```
    SENTENCE=T-EST ID1=5
DEFINE C2T6 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=TEMP  &
    SENTENCE=T-EST ID1=6
DEFINE C2T7 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=TEMP  &
    SENTENCE=T-EST ID1=7
DEFINE C2T8 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=TEMP  &
    SENTENCE=T-EST ID1=8
DEFINE C2T9 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=TEMP  &
    SENTENCE=T-EST ID1=9
DEFINE C2T10 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=TEMP  &
    SENTENCE=T-EST ID1=10
DEFINE C2T11 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=TEMP  &
    SENTENCE=T-EST ID1=11
DEFINE C2T12 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=TEMP  &
    SENTENCE=T-EST ID1=12
DEFINE C2T13 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=TEMP  &
    SENTENCE=T-EST ID1=13
DEFINE C2T14 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=TEMP  &
    SENTENCE=T-EST ID1=14
DEFINE C2T15 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=TEMP  &
    SENTENCE=T-EST ID1=15
DEFINE C2T16 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=TEMP  &
    SENTENCE=T-EST ID1=16
DEFINE C2T17 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=TEMP  &
    SENTENCE=T-EST ID1=17
DEFINE C2T18 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=TEMP  &
    SENTENCE=T-EST ID1=18
DEFINE C2T19 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=TEMP  &
    SENTENCE=T-EST ID1=19
DEFINE C2T20 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=TEMP  &
    SENTENCE=T-EST ID1=20
DEFINE C2T21 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=TEMP  &
    SENTENCE=T-EST ID1=21
DEFINE C2T22 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=TEMP  &
    SENTENCE=T-EST ID1=22
DEFINE C2T23 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=TEMP  &
    SENTENCE=T-EST ID1=23
DEFINE C2T24 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=TEMP  &
    SENTENCE=T-EST ID1=24
DEFINE C4T1 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=TEMP  &
    SENTENCE=STR-T-EST ID1=C4 ID2=1
DEFINE C4T2 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=TEMP  &
    SENTENCE=STR-T-EST ID1=C4 ID2=2
DEFINE C4T3 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=TEMP  &
    SENTENCE=STR-T-EST ID1=C4 ID2=3
```

```
DEFINE C5T1 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=TEMP  &
    SENTENCE=STR-T-EST ID1=C5 ID2=1
DEFINE C5T2 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=TEMP  &
    SENTENCE=STR-T-EST ID1=C5 ID2=2
DEFINE C5T3 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=TEMP  &
    SENTENCE=STR-T-EST ID1=C5 ID2=3
DEFINE C1L1 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=L-EST ID1=1
DEFINE C1L2 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=L-EST ID1=2
DEFINE C1L3 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=L-EST ID1=3
DEFINE C1L4 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=L-EST ID1=4
DEFINE C1L5 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=L-EST ID1=5
DEFINE C1L6 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=L-EST ID1=6
DEFINE C1L7 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=L-EST ID1=7
DEFINE C1L8 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=L-EST ID1=8
DEFINE C1L9 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=L-EST ID1=9
DEFINE C1L10 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=L-EST ID1=10
DEFINE C1L11 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=L-EST ID1=11
DEFINE C1L12 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=L-EST ID1=12
DEFINE C1L13 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=L-EST ID1=13
DEFINE C1L14 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=L-EST ID1=14
DEFINE C1L15 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=L-EST ID1=15
DEFINE C1L16 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=L-EST ID1=16
DEFINE C1L17 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=L-EST ID1=17
DEFINE C1L18 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=L-EST ID1=18
DEFINE C1L19 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=L-EST ID1=19
DEFINE C1L20 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
```

```
    SENTENCE=L-EST ID1=20
DEFINE C1L21 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=L-EST ID1=21
DEFINE C1L22 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=L-EST ID1=22
DEFINE C1L23 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=L-EST ID1=23
DEFINE C1L24 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=L-EST ID1=24
DEFINE C1L25 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=L-EST ID1=25
DEFINE C1L26 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=L-EST ID1=26
DEFINE C1L27 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=L-EST ID1=27
DEFINE C1L28 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=L-EST ID1=28
DEFINE C1L29 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=L-EST ID1=29
DEFINE C1V1 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=V-EST ID1=1
DEFINE C1V2 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=V-EST ID1=2
DEFINE C1V3 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=V-EST ID1=3
DEFINE C1V4 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=V-EST ID1=4
DEFINE C1V5 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=V-EST ID1=5
DEFINE C1V6 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=V-EST ID1=6
DEFINE C1V7 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=V-EST ID1=7
DEFINE C1V8 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=V-EST ID1=8
DEFINE C1V9 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=V-EST ID1=9
DEFINE C1V10 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=V-EST ID1=10
DEFINE C1V11 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=V-EST ID1=11
DEFINE C1V12 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=V-EST ID1=12
DEFINE C1V13 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=V-EST ID1=13
```

```
DEFINE C1V14 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=V-EST ID1=14
DEFINE C1V15 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=V-EST ID1=15
DEFINE C1V16 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=V-EST ID1=16
DEFINE C1V17 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=V-EST ID1=17
DEFINE C1V18 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=V-EST ID1=18
DEFINE C1V19 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=V-EST ID1=19
DEFINE C1V20 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=V-EST ID1=20
DEFINE C1V21 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=V-EST ID1=21
DEFINE C1V22 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=V-EST ID1=22
DEFINE C1V23 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=V-EST ID1=23
DEFINE C1V24 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=V-EST ID1=24
DEFINE C1V25 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=V-EST ID1=25
DEFINE C1V26 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=V-EST ID1=26
DEFINE C1V27 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=V-EST ID1=27
DEFINE C1V28 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=V-EST ID1=28
DEFINE C1V29 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=V-EST ID1=29
DEFINE C2L1 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
    SENTENCE=L-EST ID1=1
DEFINE C2L2 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
    SENTENCE=L-EST ID1=2
DEFINE C2L3 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
    SENTENCE=L-EST ID1=3
DEFINE C2L4 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
    SENTENCE=L-EST ID1=4
DEFINE C2L5 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
    SENTENCE=L-EST ID1=5
DEFINE C2L6 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
    SENTENCE=L-EST ID1=6
DEFINE C2L7 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
```

```
    SENTENCE=L-EST ID1=7
DEFINE C2L8 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
    SENTENCE=L-EST ID1=8
DEFINE C2L9 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
    SENTENCE=L-EST ID1=9
DEFINE C2L10 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
    SENTENCE=L-EST ID1=10
DEFINE C2L11 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
    SENTENCE=L-EST ID1=11
DEFINE C2L12 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
    SENTENCE=L-EST ID1=12
DEFINE C2L13 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
    SENTENCE=L-EST ID1=13
DEFINE C2L14 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
    SENTENCE=L-EST ID1=14
DEFINE C2L15 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
    SENTENCE=L-EST ID1=15
DEFINE C2L16 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
    SENTENCE=L-EST ID1=16
DEFINE C2L17 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
    SENTENCE=L-EST ID1=17
DEFINE C2L18 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
    SENTENCE=L-EST ID1=18
DEFINE C2L19 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
    SENTENCE=L-EST ID1=19
DEFINE C2L20 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
    SENTENCE=L-EST ID1=20
DEFINE C2L21 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
    SENTENCE=L-EST ID1=21
DEFINE C2L22 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
    SENTENCE=L-EST ID1=22
DEFINE C2L23 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
    SENTENCE=L-EST ID1=23
DEFINE C2L24 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
    SENTENCE=L-EST ID1=24
DEFINE C2V1 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
    SENTENCE=V-EST ID1=1
DEFINE C2V2 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
    SENTENCE=V-EST ID1=2
DEFINE C2V3 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
    SENTENCE=V-EST ID1=3
DEFINE C2V4 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
    SENTENCE=V-EST ID1=4
DEFINE C2V5 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
    SENTENCE=V-EST ID1=5
```

```
DEFINE C2V6 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
    SENTENCE=V-EST ID1=6
DEFINE C2V7 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
    SENTENCE=V-EST ID1=7
DEFINE C2V8 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
    SENTENCE=V-EST ID1=8
DEFINE C2V9 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
    SENTENCE=V-EST ID1=9
DEFINE C2V10 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
    SENTENCE=V-EST ID1=10
DEFINE C2V11 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
    SENTENCE=V-EST ID1=11
DEFINE C2V12 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
    SENTENCE=V-EST ID1=12
DEFINE C2V13 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
    SENTENCE=V-EST ID1=13
DEFINE C2V14 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
    SENTENCE=V-EST ID1=14
DEFINE C2V15 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
    SENTENCE=V-EST ID1=15
DEFINE C2V16 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
    SENTENCE=V-EST ID1=16
DEFINE C2V17 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
    SENTENCE=V-EST ID1=17
DEFINE C2V18 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
    SENTENCE=V-EST ID1=18
DEFINE C2V19 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
    SENTENCE=V-EST ID1=19
DEFINE C2V20 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
    SENTENCE=V-EST ID1=20
DEFINE C2V21 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
    SENTENCE=V-EST ID1=21
DEFINE C2V22 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
    SENTENCE=V-EST ID1=22
DEFINE C2V23 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
    SENTENCE=V-EST ID1=23
DEFINE C2V24 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
    SENTENCE=V-EST ID1=24
DEFINE C4L1 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
    SENTENCE=STR-L-EST ID1=C4 ID2=1
DEFINE C4L2 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
    SENTENCE=STR-L-EST ID1=C4 ID2=2
DEFINE C4L3 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
    SENTENCE=STR-L-EST ID1=C4 ID2=3
DEFINE C5L1 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
```

```
        SENTENCE=STR-L-EST ID1=C5 ID2=1
    DEFINE C5L2 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
        SENTENCE=STR-L-EST ID1=C5 ID2=2
    DEFINE C5L3 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
        SENTENCE=STR-L-EST ID1=C5 ID2=3
    DEFINE C4V1 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
        SENTENCE=STR-V-EST ID1=C4 ID2=1
    DEFINE C4V2 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
        SENTENCE=STR-V-EST ID1=C4 ID2=2
    DEFINE C4V3 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
        SENTENCE=STR-V-EST ID1=C4 ID2=3
    DEFINE C5V1 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
        SENTENCE=STR-V-EST ID1=C5 ID2=1
    DEFINE C5V2 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
        SENTENCE=STR-V-EST ID1=C5 ID2=2
    DEFINE C5V3 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
        SENTENCE=STR-V-EST ID1=C5 ID2=3
F     WRITE(NTERM,*) '+++ DATA RECONCILITAION FORTRAN STARTED +++'
C READ IN MEASURED VALUES FROM FILE
F     INQUIRE (FILE='IMEAS.DAT', EXIST=THERE)
F     IF (.NOT.THERE) THEN
F       WRITE(NTERM,*) '*** ERROR: IMEAS.DAT DOES NOT EXIST ***'
F       WRITE(NRPT,*) '*** ERROR: IMEAS.DAT WAS NOT FOUND ***'
F       WRITE(NHSTRY,*) '*** ERROR: IMEAS.DAT DID NOT EXIST ***'
F       GO TO 50
F     ENDIF
F     OPEN (UNIT=63, STATUS='OLD', FILE='IMEAS.DAT')
F     REWIND 63
F     READ(63,100) SF01
F     READ(63,100) SF11
F     READ(63,100) SF12
F     READ(63,100) SF13
F     READ(63,100) SF21
F     READ(63,100) SF22
F     READ(63,100) SF31
F     READ(63,100) SF32
F     READ(63,100) SF41
F     READ(63,100) SF42
F     READ(63,100) SF51
F     READ(63,100) SF52
F     READ(63,100) SF61
F     READ(63,100) SF62
F     READ(63,100) ST01
F     READ(63,100) ST12
F     READ(63,100) ST13
```

```
F     READ(63,100) ST21
F     READ(63,100) ST22
F     READ(63,100) ST23
F     READ(63,100) ST31
F     READ(63,10C) ST32
F     READ(63,100) ST41
F     READ(63,100) ST42
F     READ(63,100) ST51
F     READ(63,100) ST52
F     READ(63,100) ST61
F     READ(63,100) ST62
F     CLOSE (UNIT=63)
F     WRITE(NTERM,*) 'IMEAS.DAT READ SUCCESSFULLY'
C READ VARIANCES FOR MEASURED VARAIABLES FROM FILE
F  50 INQUIRE (FILE='IVARNC.DAT', EXIST=THERE)
F     IF (.NOT.THERE) THEN
F       WRITE(NTERM,*) '*** ERROR: IVARNC.DAT DOES NOT EXIST ***'
F       WRITE(NRPT,*) '*** ERROR: IVARNC.DAT WAS NOT FOUND ***'
F       WRITE(NHSTRY,*) '*** ERROR: IVARNC.DAT DID NOT EXIST ***'
F       GO TO 52
F     ENDIF
F     OPEN (UNIT=63, STATUS='OLD', FILE='IVARNC.DAT')
F     REWIND 63
F     READ(63,100) VF01
F     READ(63,100) VF11
F     READ(63,100) VF12
F     READ(63,100) VF13
F     READ(63,100) VF21
F     READ(63,100) VF22
F     READ(63,100) VF31
F     READ(63,100) VF32
F     READ(63,100) VF41
F     READ(63,100) VF42
F     READ(63,100) VF51
F     READ(63,100) VF52
F     READ(63,100) VF61
F     READ(63,100) VF62
F     READ(63,100) VT01
F     READ(63,100) VT12
F     READ(63,100) VT13
F     READ(63,100) VT21
F     READ(63,100) VT22
F     READ(63,100) VT23
F     READ(63,100) VT31
F     READ(63,100) VT32
```

```
F     READ(63,100) VT41
F     READ(63,100) VT42
F     READ(63,100) VT51
F     READ(63,100) VT52
F     READ(63,100) VT61
F     READ(63,100) VT62
F     CLOSE (UNIT=63)
F     WRITE(NTERM,*) 'IVARNC.DAT READ SUCCESSFULLY'
C READ DATA RECONCILIATION WEIGHTS FOR EACH VARIABLE FROM FILE
F  52 INQUIRE (FILE='IWGHT.DAT', EXIST=THERE)
F     IF (.NOT.THERE) THEN
F       WRITE(NTERM,*) '*** ERROR: IWGHT.DAT DOES NOT EXIST ***'
F       WRITE(NRPT,*) '*** ERROR: IWGHT.DAT WAS NOT FOUND ***'
F       WRITE(NHSTRY,*) '*** ERROR: IWGHT.DAT DID NOT EXIST ***'
F       GO TO 54
F     ENDIF
F     OPEN (UNIT=63, STATUS='OLD', FILE='IWGHT.DAT')
F     REWIND 63
F     READ(63,100) WF01
F     READ(63,100) WF11
F     READ(63,100) WF12
F     READ(63,100) WF13
F     READ(63,100) WF21
F     READ(63,100) WF22
F     READ(63,100) WF31
F     READ(63,100) WF32
F     READ(63,100) WF41
F     READ(63,100) WF42
F     READ(63,100) WF51
F     READ(63,100) WF52
F     READ(63,100) WF61
F     READ(63,100) WF62
F     READ(63,100) WT01
F     READ(63,100) WT12
F     READ(63,100) WT13
F     READ(63,100) WT21
F     READ(63,100) WT22
F     READ(63,100) WT23
F     READ(63,100) WT31
F     READ(63,100) WT32
F     READ(63,100) WT41
F     READ(63,100) WT42
F     READ(63,100) WT51
F     READ(63,100) WT52
F     READ(63,100) WT61
```

```
F     READ(63,100) WT62
F     CLOSE (UNIT=63)
F     WRITE(NTERM,*) 'IDWGHT.DAT READ SUCCESSFULLY'
C READ DATA FREE VARIABLE BOUNDS FOR DATA RECONCILIATION
F  54 INQUIRE (FILE='IDFREE.DAT', EXIST=THERE)
F     IF (.NOT.THERE) THEN
F       WRITE(NTERM,*) '*** ERROR: IDFREE.DAT DOES NOT EXIST ***'
F       WRITE(NRPT,*) '*** ERROR: IDFREE.DAT WAS NOT FOUND ***'
F       WRITE(NHSTRY,*) '*** ERROR: IDFREE.DAT DID NOT EXIST ***'
F       GO TO 56
F     ENDIF
F     OPEN (UNIT=63, STATUS='OLD', FILE='IDFREE.DAT')
F     REWIND 63
F     READ(63,100) BSTABF
F     READ(63,100) BSTABT
F     READ(63,100) BC1RR
F     READ(63,100) BC1RDV
F     READ(63,100) BC1RBF
F     READ(63,100) BC1RBT
F     READ(63,100) BE1T
F     READ(63,100) BC2MD
F     READ(63,100) BC2RBF
F     READ(63,100) BC2RBT
F     READ(63,100) BC2SCT
F     READ(63,100) BC4QN
F     READ(63,100) BC5QN
F     READ(63,100) BC2C4F
F     READ(63,100) BC2C5F
F     CLOSE (UNIT=63)
F     WRITE(NTERM,*) 'IDFREE.DAT READ SUCCESSFULLY'
C READ PLANT SPECIFICATIONS FROM FILE
F  56 INQUIRE (FILE='ISMST.DAT', EXIST=THERE)
F     IF (.NOT.THERE) THEN
F       WRITE(NTERM,*) '*** ERROR: ISMST.DAT DOES NOT EXIST ***'
F       WRITE(NRPT,*) '*** ERROR: ISMST.DAT WAS NOT FOUND ***'
F       WRITE(NHSTRY,*) '*** ERROR: ISMST.DAT DID NOT EXIST ***'
F       GO TO 58
F     END IF
F     OPEN (UNIT=63, STATUS='OLD', FILE='ISMST.DAT')
C   STAB STREAM SPECIFICATIONS
F     READ(63,100) STABF
F     READ(63,100) STABT
F     READ(63,100) STABP
F     READ(63,100) STBZ1
F     READ(63,100) STBZ2
```

```
F     READ(63,100) STBZ3
F     READ(63,100) STBZ4
F     READ(63,100) STBZ5
F     READ(63,100) STBZ6
F     READ(63,100) STBZ7
F     READ(63,100) STBZ8
F     READ(63,100) STBZ9
F     READ(63,100) STBZ10
F     READ(63,100) STBZ11
F     READ(63,100) STBZ12
F     READ(63,100) STBZ13
F     READ(63,100) STBZ14
F     READ(63,100) STBZ15
F     READ(63,100) STBZ16
F     READ(63,100) STBZ17
F     READ(63,100) STBZ18
F     READ(63,100) STBZ19
F     READ(63,100) STBZ20
F     READ(63,100) STBZ21
F     READ(63,100) STBZ22
F     READ(63,100) STBZ23
C   C2REBL STREAM ESTIMATIONS
F     READ(63,100) C2RBF1
F     C2RBF2 = C2RBF1
F     READ(63,100) C2RBT1
F     C2RBT2 = C2RBT1
F     READ(63,100) C2RBP1
F     C2RBP2 = C2RBP1
F     READ(63,100) C2RZ1
F     READ(63,100) C2RZ2
F     READ(63,100) C2RZ3
F     READ(63,100) C2RZ4
F     READ(63,100) C2RZ5
F     READ(63,100) C2RZ6
F     READ(63,100) C2RZ7
F     READ(63,100) C2RZ8
F     READ(63,100) C2RZ9
F     READ(63,100) C2RZ10
F     READ(63,100) C2RZ11
F     READ(63,100) C2RZ12
F     READ(63,100) C2RZ13
F     READ(63,100) C2RZ14
F     READ(63,100) C2RZ15
F     READ(63,100) C2RZ16
F     READ(63,100) C2RZ17
```

```
F     READ(63,100) C2RZ18
F     READ(63,100) C2RZ19
F     READ(63,100) C2RZ20
F     READ(63,100) C2RZ21
F     READ(63,100) C2RZ22
F     READ(63,100) C2RZ23
C   E1 HEATER SPECIFICATIONS
F     READ(63,100) E1T
F     READ(63,100) E1P
C   C1 COLUMN SPECIFICATIONS
F     READ(63,100) C1RR
F     READ(63,100) C1RDV
F     READ(63,100) C1RBFL
F     READ(63,100) C1RBT
F     READ(63,100) C1DP
F     READ(63,100) C1P1
F     READ(63,100) C1P2
C   C2 COLUMN SPECIFICATIONS
F     READ(63,100) C2MD
F     READ(63,100) C2SC
F     READ(63,100) C2DP
F     READ(63,100) C2P1
F     READ(63,100) C2P2
F     READ(63,100) C2IC1
F     READ(63,100) C2IC2
C   C4 COLUMN SPECIFICATIONS
F     READ(63,100) C4QN
C   C5 COLUMN SPECIFICATIONS
F     READ(63,100) C5QN
C   C1 TEMPERATURE PROFILE
F     READ(63,100) C1T1
F     READ(63,100) C1T2
F     READ(63,100) C1T3
F     READ(63,100) C1T4
F     READ(63,100) C1T5
F     READ(63,100) C1T6
F     READ(63,100) C1T7
F     READ(63,100) C1T8
F     READ(63,100) C1T9
F     READ(63,100) C1T10
F     READ(63,100) C1T11
F     READ(63,100) C1T12
F     READ(63,100) C1T13
F     READ(63,100) C1T14
F     READ(63,100) C1T15
```

```
F     READ(63,100) C1T16
F     READ(63,100) C1T17
F     READ(63,100) C1T18
F     READ(63,100) C1T19
F     READ(63,100) C1T20
F     READ(63,100) C1T21
F     READ(63,100) C1T22
F     READ(63,100) C1T23
F     READ(63,100) C1T24
F     READ(63,100) C1T25
F     READ(63,100) C1T26
F     READ(63,100) C1T27
F     READ(63,100) C1T28
F     READ(63,100) C1T29
C   C2 TEMPERATURE PROFILE
F     READ(63,100) C2T1
F     READ(63,100) C2T2
F     READ(63,100) C2T3
F     READ(63,100) C2T4
F     READ(63,100) C2T5
F     READ(63,100) C2T6
F     READ(63,100) C2T7
F     READ(63,100) C2T8
F     READ(63,100) C2T9
F     READ(63,100) C2T10
F     READ(63,100) C2T11
F     READ(63,100) C2T12
F     READ(63,100) C2T13
F     READ(63,100) C2T14
F     READ(63,100) C2T15
F     READ(63,100) C2T16
F     READ(63,100) C2T17
F     READ(63,100) C2T18
F     READ(63,100) C2T19
F     READ(63,100) C2T20
F     READ(63,100) C2T21
F     READ(63,100) C2T22
F     READ(63,100) C2T23
F     READ(63,100) C2T24
C   C4 TEMPERATURE PROFILE
F     READ(63,100) C4T1
F     READ(63,100) C4T2
F     READ(63,100) C4T3
C   C5 TEMPERATURE PROFILE
F     READ(63,100) C5T1
```

```
F     READ(63,100) C5T2
F     READ(63,100) C5T3
C   C1 LIQUID AND VAPOUR FLOW PROFILES
F     READ(63,100) C1L1
F     READ(63,100) C1L2
F     READ(63,100) C1L3
F     READ(63,100) C1L4
F     READ(63,100) C1L5
F     READ(63,100) C1L6
F     READ(63,100) C1L7
F     READ(63,100) C1L8
F     READ(63,100) C1L9
F     READ(63,100) C1L10
F     READ(63,100) C1L11
F     READ(63,100) C1L12
F     READ(63,100) C1L13
F     READ(63,100) C1L14
F     READ(63,100) C1L15
F     READ(63,100) C1L16
F     READ(63,100) C1L17
F     READ(63,100) C1L18
F     READ(63,100) C1L19
F     READ(63,100) C1L20
F     READ(63,100) C1L21
F     READ(63,100) C1L22
F     READ(63,100) C1L23
F     READ(63,100) C1L24
F     READ(63,100) C1L25
F     READ(63,100) C1L26
F     READ(63,100) C1L27
F     READ(63,100) C1L28
F     READ(63,100) C1L29
F     READ(63,100) C1V1
F     READ(63,100) C1V2
F     READ(63,100) C1V3
F     READ(63,100) C1V4
F     READ(63,100) C1V5
F     READ(63,100) C1V6
F     READ(63,100) C1V7
F     READ(63,100) C1V8
F     READ(63,100) C1V9
F     READ(63,100) C1V10
F     READ(63,100) C1V11
F     READ(63,100) C1V12
F     READ(63,100) C1V13
```

```
F     READ(63,100) C1V14
F     READ(63,100) C1V15
F     READ(63,100) C1V16
F     READ(63,100) C1V17
F     READ(63,100) C1V18
F     READ(63,100) C1V19
F     READ(63,100) C1V20
F     READ(63,100) C1V21
F     READ(63,100) C1V22
F     READ(63,100) C1V23
F     READ(63,100) C1V24
F     READ(63,100) C1V25
F     READ(63,100) C1V26
F     READ(63,100) C1V27
F     READ(63,100) C1V28
F     READ(63,100) C1V29
C   C2 LIQUID AND VAPOUR FLOW PROFILES
F     READ(63,100) C2L1
F     READ(63,100) C2L2
F     READ(63,100) C2L3
F     READ(63,100) C2L4
F     READ(63,100) C2L5
F     READ(63,100) C2L6
F     READ(63,100) C2L7
F     READ(63,100) C2L8
F     READ(63,100) C2L9
F     READ(63,100) C2L10
F     READ(63,100) C2L11
F     READ(63,100) C2L12
F     READ(63,100) C2L13
F     READ(63,100) C2L14
F     READ(63,100) C2L15
F     READ(63,100) C2L16
F     READ(63,100) C2L17
F     READ(63,100) C2L18
F     READ(63,100) C2L19
F     READ(63,100) C2L20
F     READ(63,100) C2L21
F     READ(63,100) C2L22
F     READ(63,100) C2L23
F     READ(63,100) C2L24
F     READ(63,100) C2V1
F     READ(63,100) C2V2
F     READ(63,100) C2V3
F     READ(63,100) C2V4
```

```
F     READ(63,100) C2V5
F     READ(63,100) C2V6
F     READ(63,100) C2V7
F     READ(63,100) C2V8
F     READ(63,100) C2V9
F     READ(63,100) C2V10
F     READ(63,100) C2V11
F     READ(63,100) C2V12
F     READ(63,100) C2V13
F     READ(63,100) C2V14
F     READ(63,100) C2V15
F     READ(63,100) C2V16
F     READ(63,100) C2V17
F     READ(63,100) C2V18
F     READ(63,100) C2V19
F     READ(63,100) C2V20
F     READ(63,100) C2V21
F     READ(63,100) C2V22
F     READ(63,100) C2V23
F     READ(63,100) C2V24
C   C4 LIQUID AND VAPOUR FLOW PROFILES
F     READ(63,100) C4L1
F     READ(63,100) C4L2
F     READ(63,100) C4L3
F     READ(63,100) C4V1
F     READ(63,100) C4V2
F     READ(63,100) C4V3
F     READ(63,100) C5L1
F     READ(63,100) C5L2
F     READ(63,100) C5L3
F     READ(63,100) C5V1
F     READ(63,100) C5V2
F     READ(63,100) C5V3
F     CLOSE (UNIT=63)
F     WRITE(NTERM,*) 'ISMST.DAT READ SUCCESSFULLY'
F  58 WRITE(NTERM,*) '+++ DATA RECONCILIATION FORTRAN COMPLETE +++'
C FORMAT STATEMENTS
F 100 FORMAT(15X, E15.8)
    EXECUTE FIRST

FORTRAN OPDATA
F     COMMON / USRPVL / PVAP,PDIST,PLTISO,PHVISO,PLTDST,PSSFED,PVALS
F     COMMON / USRCST / CSTAB,COF1,COF2
F     COMMON / USRQLT / QLTISO
F     LOGICAL THERE
```

```
DEFINE F01 STREAM-VAR STREAM=STAB SUBSTREAM=MIXED  &
    VARIABLE=STDVOL-FLOW
DEFINE F11 STREAM-PROP STREAM=VAP PROPERTY=V-STDVOL
DEFINE F11STD STREAM-VAR STREAM=VAP SUBSTREAM=MIXED  &
    VARIABLE=STDVOL-FLOW
DEFINE F12 STREAM-VAR STREAM=C1RFLX SUBSTREAM=MIXED  &
    VARIABLE=STDVOL-FLOW
DEFINE F13 STREAM-VAR STREAM=DIST-C4 SUBSTREAM=MIXED  &
    VARIABLE=STDVOL-FLOW
DEFINE F21 STREAM-VAR STREAM=C1REBL SUBSTREAM=MIXED  &
    VARIABLE=STDVOL-FLOW
DEFINE F22 STREAM-VAR STREAM=SPLTFEED SUBSTREAM=MIXED  &
    VARIABLE=STDVOL-FLOW
DEFINE F31 STREAM-VAR STREAM=LT-ISO SUBSTREAM=MIXED  &
    VARIABLE=STDVOL-FLOW
DEFINE F32 STREAM-VAR STREAM=C2RFLX SUBSTREAM=MIXED  &
    VARIABLE=STDVOL-FLOW
DEFINE F41 STREAM-VAR STREAM=IC1 SUBSTREAM=MIXED  &
    VARIABLE=STDVOL-FLOW
DEFINE F42 STREAM-VAR STREAM=HV-ISO SUBSTREAM=MIXED  &
    VARIABLE=STDVOL-FLOW
DEFINE F51 STREAM-VAR STREAM=IC2 SUBSTREAM=MIXED  &
    VARIABLE=STDVOL-FLOW
DEFINE F52 STREAM-VAR STREAM=LT-DIST SUBSTREAM=MIXED  &
    VARIABLE=STDVOL-FLOW
DEFINE F61 STREAM-VAR STREAM=C2REBL SUBSTREAM=MIXED  &
    VARIABLE=STDVOL-FLOW
DEFINE F62 STREAM-VAR STREAM=SSFEED SUBSTREAM=MIXED  &
    VARIABLE=STDVOL-FLOW
DEFINE T01 STREAM-VAR STREAM=STAB SUBSTREAM=MIXED  &
    VARIABLE=TEMP
DEFINE T11 BLOCK-VAR BLOCK=C1 VARIABLE=TEMP  &
    SENTENCE=PROFILE ID1=3
DEFINE T12 BLOCK-VAR BLOCK=C1 VARIABLE=TEMP  &
    SENTENCE=PROFILE ID1=2
DEFINE T13 STREAM-VAR STREAM=DIST-C4 SUBSTREAM=MIXED  &
    VARIABLE=TEMP
DEFINE T21 STREAM-VAR STREAM=SPLTFEED SUBSTREAM=MIXED  &
    VARIABLE=TEMP
DEFINE T22 STREAM-VAR STREAM=C1REBL SUBSTREAM=MIXED  &
    VARIABLE=TEMP
DEFINE T23 STREAM-VAR STREAM=SPLT SUBSTREAM=MIXED  &
    VARIABLE=TEMP
DEFINE T31 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=TEMP  &
    SENTENCE=PROFILE ID1=2
```

```
DEFINE T32 STREAM-VAR STREAM=LT-ISO SUBSTREAM=MIXED  &
    VARIABLE=TEMP
DEFINE T41 STREAM-VAR STREAM=IC1 SUBSTREAM=MIXED  &
    VARIABLE=TEMP
DEFINE T42 STREAM-VAR STREAM=HV-ISO SUBSTREAM=MIXED  &
    VARIABLE=TEMP
DEFINE T51 STREAM-VAR STREAM=IC2 SUBSTREAM=MIXED  &
    VARIABLE=TEMP
DEFINE T52 STREAM-VAR STREAM=LT-DIST SUBSTREAM=MIXED  &
    VARIABLE=TEMP
DEFINE T61 STREAM-VAR STREAM=SSFEED SUBSTREAM=MIXED  &
    VARIABLE=TEMP
DEFINE T62 STREAM-VAR STREAM=C2REBL SUBSTREAM=MIXED  &
    VARIABLE=TEMP
DEFINE P01 STREAM-VAR STREAM=STAB SUBSTREAM=MIXED  &
    VARIABLE=PRES
DEFINE DCIC5 MOLE-FRAC STREAM=DIST-C4 SUBSTREAM=MIXED  &
    COMPONENT=IPENTANE
DEFINE DCNC5 MOLE-FRAC STREAM=DIST-C4 SUBSTREAM=MIXED  &
    COMPONENT=PENTANE
DEFINE RVA31 STREAM-PROP STREAM=LT-ISO PROPERTY=RVP
DEFINE TBUB31 STREAM-PROP STREAM=LT-ISO PROPERTY=TBUB
DEFINE D1A41 STREAM-PROP STREAM=HV-ISO PROPERTY=TBP-10
DEFINE D9A41 STREAM-PROP STREAM=HV-ISO PROPERTY=TBP-90
DEFINE D1A51 STREAM-PROP STREAM=LT-DIST PROPERTY=TBP-10
DEFINE D9A51 STREAM-PROP STREAM=LT-DIST PROPERTY=TBP-90
DEFINE C1QN BLOCK-VAR BLOCK=C1 VARIABLE=DUTY  &
    SENTENCE=PA-RESULTS ID1=C1REBL
DEFINE C2QN BLOCK-VAR BLOCK=F2 VARIABLE=QCALC  &
    SENTENCE=PARAM
DEFINE STABF STREAM-VAR STREAM=STAB SUBSTREAM=MIXED  &
    VARIABLE=MOLE-FLOW
DEFINE STABT STREAM-VAR STREAM=STAB SUBSTREAM=MIXED  &
    VARIABLE=TEMP
DEFINE STABP STREAM-VAR STREAM=STAB SUBSTREAM=MIXED  &
    VARIABLE=PRES
VECTOR-DEF STABZ STREAM STAB
DEFINE C2RBLF STREAM-VAR STREAM=C2REBL SUBSTREAM=MIXED  &
    VARIABLE=MOLE-FLOW
DEFINE C2RBLT STREAM-VAR STREAM=C2REBL SUBSTREAM=MIXED  &
    VARIABLE=TEMP
DEFINE C2RBLP STREAM-VAR STREAM=C2REBL SUBSTREAM=MIXED  &
    VARIABLE=PRES
VECTOR-DEF C2RBLZ STREAM C2REBL
DEFINE E1T BLOCK-VAR BLOCK=E1 VARIABLE=TEMP SENTENCE=PARAM
```

```
DEFINE E1P BLOCK-VAR BLOCK=E1 VARIABLE=PRES SENTENCE=PARAM
DEFINE C1RR BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-RR  &
    SENTENCE=COL-SPECS
DEFINE C1RDV BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-RDV  &
    SENTENCE=COL-SPECS
DEFINE C1RBFL BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=PUMPAROUND ID1=C1REBL
DEFINE C1RBT BLOCK-VAR BLOCK=C1 VARIABLE=TEMP  &
    SENTENCE=PUMPAROUND ID1=C1REBL
DEFINE C1P1 BLOCK-VAR BLOCK=C1 VARIABLE=PRES  &
    SENTENCE=P-SPEC ID1=1
DEFINE C1P2 BLOCK-VAR BLOCK=C1 VARIABLE=PRES  &
    SENTENCE=P-SPEC ID1=2
DEFINE C1DP BLOCK-VAR BLOCK=C1 VARIABLE=DP-COL  &
    SENTENCE=COL-SPECS
VECTOR-DEF C1TEMP PROFILE BLOCK=C1 VARIABLE=TEMP  &
    SENTENCE=PROFILE
DEFINE C2MD BLOCK-VAR BLOCK=C2C4C5 VARIABLE=MOLE-D  &
    SENTENCE=COL-SPECS
DEFINE C2SC BLOCK-VAR BLOCK=C2C4C5 VARIABLE=T1  &
    SENTENCE=COL-SPECS
DEFINE C2P1 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=PRES  &
    SENTENCE=P-SPEC ID1=1
DEFINE C2P2 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=PRES  &
    SENTENCE=P-SPEC ID1=2
DEFINE C2DP BLOCK-VAR BLOCK=C2C4C5 VARIABLE=DP-COL  &
    SENTENCE=COL-SPECS
VECTOR-DEF C2TEMP PROFILE BLOCK=C2C4C5 VARIABLE=TEMP  &
    SENTENCE=PROFILE
VECTOR-DEF C4C5T PROFILE BLOCK=C2C4C5 VARIABLE=TEMP  &
    SENTENCE=STR-PROFILE
DEFINE C4QN BLOCK-VAR BLOCK=C2C4C5 VARIABLE=Q-REB  &
    SENTENCE=STRIPPER ID1=C4
DEFINE C5QN BLOCK-VAR BLOCK=C2C4C5 VARIABLE=Q-REB  &
    SENTENCE=STRIPPER ID1=C5
DEFINE C2IC1 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=MOLE-DRAW  &
    SENTENCE=STRIPPER ID1=C4
DEFINE C2IC2 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=MOLE-DRAW  &
    SENTENCE=STRIPPER ID1=C5
VECTOR-DEF C1LEST PROFILE BLOCK=C1 VARIABLE=LRATE  &
    SENTENCE=PROFILE
VECTOR-DEF C1VEST PROFILE BLOCK=C1 VARIABLE=VRATE  &
    SENTENCE=PROFILE
VECTOR-DEF C2LEST PROFILE BLOCK=C2C4C5 VARIABLE=LRATE  &
    SENTENCE=PROFILE
```

```
    VECTOR-DEF C2VEST PROFILE BLOCK=C2C4C5 VARIABLE=VRATE  &
        SENTENCE=PROFILE
    VECTOR-DEF C4C5LE PROFILE BLOCK=C2C4C5 VARIABLE=LRATE  &
        SENTENCE=STR-PROFILE
    VECTOR-DEF C4C5VE PROFILE BLOCK=C2C4C5 VARIABLE=VRATE  &
        SENTENCE=STR-PROFILE
F     WRITE(NTERM,*) '+++ OPTIMIZATION RESULTS FORTRAN STARTED +++'
C WRITE TO FILE MEASURED VARIABLE VALUES AFTER OPTIMIZATION
F     INQUIRE (FILE='ROVARS.DAT', EXIST=THERE)
F     IF (THERE) THEN
F       WRITE(NTERM,*) '*** ERROR: ROVARS.DAT ALREADY EXISTS ***'
F       WRITE(NRPT,*) '*** ERROR: ROVARS.DAT ALREADY EXISTED ***'
F       WRITE(NHSTRY,*) '*** ERROR: ROVARS.DAT ALREADY EXISTS ***'
F       GO TO 50
F     ENDIF
F     OPEN (UNIT=63, STATUS='NEW', FILE='ROVARS.DAT')
F     WRITE(63,100) 'F01', F01
F     WRITE(63,100) 'F11', F11
F     WRITE(63,100) 'F12', F12
F     WRITE(63,100) 'F13', F13
F     WRITE(63,100) 'F21', F21
F     WRITE(63,100) 'F22', F22
F     WRITE(63,100) 'F31', F31
F     WRITE(63,100) 'F32', F32
F     WRITE(63,100) 'F41', F41
F     WRITE(63,100) 'F42', F42
F     WRITE(63,100) 'F51', F51
F     WRITE(63,100) 'F52', F52
F     WRITE(63,100) 'F61', F61
F     WRITE(63,100) 'F62', F62
F     WRITE(63,100) 'T01', T01
F     WRITE(63,100) 'T12', T12
F     WRITE(63,100) 'T13', T13
F     WRITE(63,100) 'T21', T21
F     WRITE(63,100) 'T22', T22
F     WRITE(63,100) 'T23', T23
F     WRITE(63,100) 'T31', T31
F     WRITE(63,100) 'T32', T32
F     WRITE(63,100) 'T41', T41
F     WRITE(63,100) 'T42', T42
F     WRITE(63,100) 'T51', T51
F     WRITE(63,100) 'T52', T52
F     WRITE(63,100) 'T61', T61
F     WRITE(63,100) 'T62', T62
F     CLOSE (UNIT=63)
```

```
F     WRITE(NTERM,*) 'ROVARS.DAT WRITTEN SUCCESSFULLY'
C WRITE TO FILE A CONSTRAINT VARIABLE SUMMARY
F  50 INQUIRE (FILE='ROCNST.DAT', EXIST=THERE)
F     IF (THERE) THEN
F       WRITE(NTERM,*) '*** ERROR: ROCNST.DAT ALREADY EXISTS ***'
F       WRITE(NRPT,*) '*** ERROR: ROCNST.DAT ALREADY EXISTED ***'
F       WRITE(NHSTRY,*) '*** ERROR: ROCNST.DAT ALREADY EXISTS ***'
F       GO TO 52
F     ENDIF
F     OPEN (UNIT=63, STATUS='NEW', FILE='ROCNST.DAT')
F     WRITE(63,100) 'F12', F12
F     WRITE(63,100) 'F32', F32
F     WRITE(63,100) 'F41', F41
F     WRITE(63,100) 'F51', F51
F     WRITE(63,100) 'T11', T11
F     WRITE(63,100) 'T22', T22
F     WRITE(63,100) 'T31', T31
F     WRITE(63,100) 'T62', T62
F     C5A11 = DCIC5 + DCNC5
F     WRITE(63,100) 'A11 (C5S)', C5A11
F     WRITE(63,100) 'A31 (RVP)', RVA31
F     WRITE(63,100) 'A31 (TBUB)', TBUB31
F     WRITE(63,100) 'A41 (TBP10)', D1A41
F     WRITE(63,100) 'A41 (TBP90)', D9A41
F     WRITE(63,100) 'A51 (TBP10)', D1A51
F     WRITE(63,100) 'A51 (TBP90)', D9A51
F     CLOSE (UNIT=63)
F     WRITE(NTERM,*) 'ROCNST.DAT WRITTEN SUCCESSFULLY'
C WRITE TO FILE OBJECTIVE FUNCTION SUMMARY
F  52 INQUIRE (FILE='ROOPFN.DAT', EXIST=THERE)
F     IF (THERE) THEN
F       WRITE(NTERM,*) '*** ERROR: ROOPFN.DAT ALREADY EXISTS ***'
F       WRITE(NRPT,*) '*** ERROR: ROOPFN.DAT ALREADY EXISTED ***'
F       WRITE(NHSTRY,*) '*** ERROR: ROOPFN.DAT ALREADY EXISTS ***'
F       GO TO 54
F     ENDIF
F     OPEN (UNIT=63, STATUS='NEW', FILE='ROOPFN.DAT')
F     PROFIT = PVALS-CSTAB-COF1-COF2
F     UCOST = COF1+COF2
F     WRITE(63,100) 'OBFUN-PROFIT', PROFIT
F     WRITE(63,100) 'TOT PROD VALS', PVALS
F     WRITE(63,100) 'TOT FEED COST', CSTAB
F     WRITE(63,100) 'TOT UTIL COST', UCOST
F     WRITE(63,100) 'VAP SVOLFLOW', F11STD
F     WRITE(63,100) 'DISTC4 SVOLFL', F13
```

```
F     WRITE(63,100) 'LTISO SVOLFL', F31
F     WRITE(63,100) 'HVISO SVOLFL', F42
F     WRITE(63,100) 'LTDIST SVOLFL', F52
F     WRITE(63,100) 'SSFEED SVOLFL', F62
F     WRITE(63,100) 'STAB SVOLFLOW', F01
F     WRITE(63,100) 'C1 REBL DUTY', C1QN
F     WRITE(63,100) 'C2 REBL DUTY', C2QN
F     WRITE(63,100) 'LT-ISO MARG', QLTISO
F     WRITE(63,100) 'LT-ISO RVP', RVA31
F     CLOSE (UNIT=63)
F     WRITE(NTERM,*) 'ROOPFN.DAT WRITTEN SUCCESSFULLY'
C WRITE TO FILE THE VARIABLES FOR SIMULATION AFTER OPTIMIZATION
F  54 INQUIRE (FILE='ROSMST.DAT', EXIST=THERE)
F     IF (THERE) THEN
F       WRITE(NTERM,*) '*** ERROR: ROSMST.DAT ALREADY EXISTS ***'
F       WRITE(NRPT,*) '*** ERROR: ROSMST.DAT ALREADY EXISTED ***'
F       WRITE(NHSTRY,*) '*** ERROR: ROSMST.DAT ALREADY EXISTS ***'
F       GO TO 56
F     ENDIF
F     OPEN (UNIT=64, STATUS='NEW', FILE='ROSMST.DAT')
C   STAB STREAM SPECIFICATIONS
F     WRITE(64,100) 'STAB MOLEFLOW', STABF
F     WRITE(64,100) 'STAB TEMP', STABT
F     WRITE(64,100) 'STAB PRESS', STABP
F     DO 12 I=1,23
F       TEMP = STABZ(I) / STABZ(24)
F       WRITE(64,105) 'STAB Z', I, TEMP
F  12 CONTINUE
C   C2REBL STREAM ESTIMATIONS
F     WRITE(64,100) 'C2REBL FLOW', C2RBLF
F     WRITE(64,100) 'C2REBL TEMP', C2RBLT
F     WRITE(64,100) 'C2REBL PRESS', C2RBLP
F     DO 14 I=1,23
F       TEMP = C2RBLZ(I) / C2RBLZ(24)
F       WRITE(64,105) 'C2REBL Z', I, TEMP
F  14 CONTINUE
C   E1 HEATER SPECIFICATIONS
F     WRITE(64,100) 'E1 TEMP', E1T
F     WRITE(64,100) 'E1 PRESS', E1P
C   C1 COLUMN SPECIFICATIONS
F     WRITE(64,100) 'C1 RFLX RATIO', C1RR
F     WRITE(64,100) 'C1 VAP/DIST', C1RDV
F     WRITE(64,100) 'C1 REBL FLOW', C1RBFL
F     WRITE(64,100) 'C1 REBL TEMP', C1RBT
F     WRITE(64,100) 'C1 COL P DROP', C1DP
```

```
F     WRITE(64,100) 'C1 TOP TRAY P', C1P1
F     WRITE(64,100) 'C1 2ND TRAY P', C1P2
C   C2 COLUMN SPECIFICATIONS
F     WRITE(64,100) 'C2 DIST FLOW', C2MD
F     WRITE(64,100) 'C2 COND SC T', C2SC
F     WRITE(64,100) 'C2 COL P DROP', C2DP
F     WRITE(64,100) 'C2 TOP TRAY P', C2P1
F     WRITE(64,100) 'C2 2ND TRAY P', C2P2
F     WRITE(64,100) 'C2-C4 IC FLOW', C2IC1
F     WRITE(64,100) 'C2-C5 IC FLOW', C2IC2
C   C4 COLUMN SPECIFICATIONS
F     WRITE(64,100) 'C4 REBL DUTY', C4QN
C   C5 COLUMN SPECIFICATIONS
F     WRITE(64,100) 'C5 REBL DUTY', C5QN
C   C1 TEMPERATURE PROFILE
F     DO 10 I=1,29
F       WRITE(64,105) 'C1 T, TRAY', I, C1TEMP(I)
F  10 CONTINUE
C   C2 TEMPERATURE PROFILE
F     DO 20 I=1,24
F       WRITE(64,105) 'C2 T, TRAY', I, C2TEMP(I)
F  20 CONTINUE
C   C4 TEMPERATURE PROFILE
F     DO 25 I=1,3
F       J = (2*I)-1
F       WRITE(64,105) 'C4 T, TRAY', I, C4C5T(J)
F  25 CONTINUE
C   C5 TEMPERATURE PROFILE
F     DO 27 I=1,3
F       J = 2*I
F       WRITE(64,105) 'C5 T, TRAY', I, C4C5T(J)
F  27 CONTINUE
C   C1 LIQUID AND VAPOUR FLOW PROFILES
F     DO 30 I=1,29
F       WRITE(64,105) 'C1 L, TRAY', I, C1LEST(I)
F  30 CONTINUE
F     DO 32 I=1,29
F       WRITE(64,105) 'C1 V, TRAY', I, C1VEST(I)
F  32 CONTINUE
C   C2 LIQUID AND VAPOUR FLOW PROFILES
F     DO 34 I=1,24
F       WRITE(64,105) 'C2 L, TRAY', I, C2LEST(I)
F  34 CONTINUE
F     DO 36 I=1,24
F       WRITE(64,105) 'C2 V, TRAY', I, C2VEST(I)
```

```
F  36 CONTINUE
F     DO 38 I=1,3
F       J = (2*I)-1
F       WRITE(64,105) 'C4 L, TRAY', I, C4C5LE(J)
F  38 CONTINUE
F     DO 40 I=1,3
F       J = (2*I)-1
F       WRITE(64,105) 'C4 V, TRAY', I, C4C5VE(J)
F  40 CONTINUE
F     DO 42 I=1,3
F       J = 2*I
F       WRITE(64,105) 'C5 L, TRAY', I, C4C5LE(J)
F  42 CONTINUE
F     DO 44 I=1,3
F       J = 2*I
F       WRITE(64,105) 'C5 V, TRAY', I, C4C5VE(J)
F  44 CONTINUE
F     CLOSE (UNIT=64)
F     WRITE(NTERM,*) 'ROSMST.DAT WRITTEN SUCCESSFULLY'
C WRITE TO FILE SET POINT VALUES
F  56 INQUIRE (FILE='ROSTPT.DAT', EXIST=THERE)
F     IF (THERE) THEN
F       WRITE(NTERM,*) '*** ERROR: ROSTPT.DAT ALREADY EXISTS ***'
F       WRITE(NRPT,*) '*** ERROR: ROSTPT.DAT ALREADY EXISTED ***'
F       WRITE(NHSTRY,*) '*** ERROR: ROSTPT.DAT ALREADY EXISTS ***'
F       GO TO 56
F     ENDIF
F     OPEN (UNIT=64, STATUS='NEW', FILE='ROSTPT.DAT')
F     WRITE(64,110) 'C1 REFLUX FLOW (F12) [BBL/DAY]', F12
F     WRITE(64,110) 'C1 TRAY 3 TEMPERATURE (T11) [F]', T11
F     WRITE(64,110) 'DIST-C4 C5 MOLE FRACTION (A11)', C5A11
F     WRITE(64,110) 'C1 REBOILER TEMPERATURE (T22) [F]', T22
F     WRITE(64,110) 'C2 REFLUX FLOW (F32) [BBL/DAY]', F32
F     WRITE(64,110) 'C4 FEED FLOW (F41) [BBL/DAY]', F41
F     WRITE(64,110) 'C5 FEED FLOW (F51) [BBL/DAY]', F51
F     WRITE(64,110) 'C2 REBOILER TEMPERATURE (T62) [F]', T62
F     WRITE(64,110) 'HV-ISO 10% TBP TEMPERATURE (A41) [F]', D1A41
F     WRITE(64,110) 'HV-ISO 90% TBP TEMPERATURE (A41) [F]', D9A41
F     WRITE(64,110) 'LT-DIST 10% TBP TEMPERATURE (A51) [F]', D1A51
F     WRITE(64,110) 'LT-DIST 90% TBP TEMPERATURE (A51) [F]', D9A51
F     CLOSE (UNIT=64)
F     WRITE(NTERM,*) 'ROSTPT.DAT WRITTEN SUCCESSFULLY'
F  58 WRITE(NTERM,*) '+++ OPTIMIZATION RESULTS FORTRAN COMPLETED +++'
C FORMAT STATEMENTS
F 100 FORMAT(A13, 2X, E15.8)
```

```
F 105 FORMAT(A10, X, I2, 2X, E15.8)
F 110 FORMAT(A40, 2X, E15.8)
    EXECUTE LAST

FORTRAN OPSPCSET
F     COMMON / USRD1 / DVAP,DDIST,DLTISO,DHVISO,DLTDST,DSSFED
F     COMMON / USRD2 / DSTAB,DFO
F     COMMON / USRPRM / PRMRVP
F     COMMON / USRSPC / SPCRVP
F     COMMON / USRCNV / CNVB2M,CNVBTM,CNVBTU,CNVP2K
F     COMMON / USRFEF / F1EFF,F2EFF
F     COMMON / USRCA1 / A11U,A11L,A31UR,A31LT,A41U1,A41L1
F     COMMON / USRCA2 / A41U9,A41L9,A51U1,A51L1,A51U9,A51L9
F     COMMON / USRCF / F12U,F12L,F32U,F41U,F51U
F     COMMON / USRCT / T11U,T11L,T22U,T22L,T31U,T31L,T62U,T62L
F     COMMON / USRCEQ / EQ1,EQ2
F     COMMON / USRP1 / PC1RR,PC1RDV,PC1RBF,PC1RBT,PE1T,PC2MD,PC2RBF
F     COMMON / USRP2 / PC2RBT,PC2SCT,PC4QN,PC5QN,PC2C4F,PC2C5F
F     LOGICAL THERE
    DEFINE C1Q1 BLOCK-VAR BLOCK=C1 VARIABLE=COND-DUTY  &
        SENTENCE=RESULTS
    DEFINE C2Q1 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=COND-DUTY  &
        SENTENCE=RESULTS
F     WRITE(NTERM,*) '+++ OPTIMIZATION FORTRAN STARTED +++'
C CONVERSIONS:
C   BBL/DAY TO M3/HR
F     CNVB2M = 0.006624471
C   BBL TO M3
F     CNVBTM = 0.1589873
C   MMBTU TO LFEBBL
F     CNVBTU = 1/6.3
C   PSIA TO KPA
F     CNVP2K = 6.8948
C EQUALITY CONSTRAINT VALUES
F     EQ1 = C1Q1
F     EQ2 = C2Q1
F     WRITE(NTERM,*) 'EQUALITY CONSTRAINTS SET SUCCESSFULLY'
C READ OPTIMIZATION OBJECTIVE FUNCTION PARAMETERS FROM FILE
F     INQUIRE (FILE='IOBJFN.DAT', EXIST=THERE)
F     IF (.NOT.THERE) THEN
F       WRITE(NTERM,*) '*** ERROR: IOBJFN.DAT DOES NOT EXIST ***'
F       WRITE(NRPT,*) '*** ERROR: IOBJFN.DAT WAS NOT FOUND ***'
F       WRITE(NHSTRY,*) '*** ERROR: IOBJFN.DAT DID NOT EXIST ***'
F       GO TO 52
F     ENDIF
```

```
F     OPEN (UNIT=63, STATUS='OLD', FILE='IOBJFN.DAT')
F     REWIND 63
F     READ(63,100) PRMRVP
F     READ(63,100) SPCRVP
F     READ(63,100) DVAP
F     READ(63,100) .DIST
F     READ(63,100) DLTISO
F     READ(63,100) DHVISO
F     READ(63,100) DLTDST
F     READ(63,100) DSSFED
F     READ(63,100) DSTAB
F     READ(63,100) DFO
F     READ(63,100) F1EFF
F     READ(63,100) F2EFF
F     CLOSE (UNIT=63)
F     WRITE(NTERM,*) 'IOBJFN.DAT READ SUCCESSFULLY'
C READ OPTIMIZATION CONSTRAINT BOUNDS FROM FILE
F  52 INQUIRE (FILE='ICNSTR.DAT', EXIST=THERE)
F     IF (.NOT.THERE) THEN
F       WRITE(NTERM,*) '*** ERROR: ICNSTR.DAT DOES NOT EXIST ***'
F       WRITE(NRPT,*) '*** ERROR: ICNSTR.DAT WAS NOT FOUND ***'
F       WRITE(NHSTRY,*) '*** ERROR: ICNSTR.DAT DID NOT EXIST ***'
F       GO TO 54
F     ENDIF
F     OPEN (UNIT=63, STATUS='OLD', FILE='ICNSTR.DAT')
F     REWIND 63
F     READ(63,100) A11U
F     READ(63,100) A11L
F     READ(63,100) A31UR
F     READ(63,100) A31LT
F     READ(63,100) A41U1
F     READ(63,100) A41L1
F     READ(63,100) A41U9
F     READ(63,100) A41L9
F     READ(63,100) A51U1
F     READ(63,100) A51L1
F     READ(63,100) A51U9
F     READ(63,100) A51L9
F     READ(63,100) F12U
F     READ(63,100) F12L
F     READ(63,100) F32U
F     READ(63,100) F41U
F     READ(63,100) F51U
F     READ(63,100) T11U
F     READ(63,100) T11L
```

```
F     READ(63,100) T22U
F     READ(63,100) T22L
F     READ(63,100) T31U
F     READ(63,100) T31L
F     READ(63,100) T62U
F     READ(63,100) T62L
F     CLOSE (UNIT=63)
F     WRITE(NTERM,*) 'ICNSTR.DAT READ SUCCESSFULLY'
C READ DATA FREE VARIABLE BOUNDS FOR DATA RECONCILIATION
F  54 INQUIRE (FILE='IOFREE.DAT', EXIST=THERE)
F     IF (.NOT.THERE) THEN
F       WRITE(NTERM,*) '*** ERROR: IOFREE.DAT DOES NOT EXIST ***'
F       WRITE(NRPT,*) '*** ERROR: IOFREE.DAT WAS NOT FOUND ***'
F       WRITE(NHSTRY,*) '*** ERROR: IOFREE.DAT DID NOT EXIST ***'
F       GO TO 56
F     ENDIF
F     OPEN (UNIT=63, STATUS='OLD', FILE='IOFREE.DAT')
F     REWIND 63
F     READ(63,100) PC1RR
F     READ(63,100) PC1RDV
F     READ(63,100) PC1RBF
F     READ(63,100) PC1RBT
F     READ(63,100) PE1T
F     READ(63,100) PC2MD
F     READ(63,100) PC2RBF
F     READ(63,100) PC2RBT
F     READ(63,100) PC2SCT
F     READ(63,100) PC4QN
F     READ(63,100) PC5QN
F     READ(63,100) PC2C4F
F     READ(63,100) PC2C5F
F     CLOSE (UNIT=63)
F     WRITE(NTERM,*) 'IOFREE.DAT READ SUCCESSFULLY'
F  56 WRITE(NTERM,*) '+++ OPTIMIZATION FORTRAN COMPLETE +++'
C FORMAT STATEMENTS
F 100 FORMAT(15X, E15.8)
F 105 FORMAT(A13, 2X, E15.8)
    EXECUTE FIRST

CONSTRAINT C5A11L
F     COMMON / USRCA1 / A11U,A11L,A31UR,A31LT,A41U1,A41L1
    DEFINE DCIC5 MOLE-FRAC STREAM=DIST-C4 SUBSTREAM=MIXED  &
        COMPONENT=IPENTANE
    DEFINE DCNC5 MOLE-FRAC STREAM=DIST-C4 SUBSTREAM=MIXED  &
        COMPONENT=PENTANE
```

```
    SPEC "DCIC5 + DCNC5" GE "A11L"
    TOL-SPEC "0.001"

CONSTRAINT C5A11U
F     COMMON / USRCA1 / A11U,A11L,A31UR,A31LT,A41U1,A41L1
    DEFINE DCIC5 MOLE-FRAC STREAM=DIST-C4 SUBSTREAM=MIXED  &
        COMPONENT=IPENTANE
    DEFINE DCNC5 MOLE-FRAC STREAM=DIST-C4 SUBSTREAM=MIXED  &
        COMPONENT=PENTANE
    SPEC "DCIC5 + DCNC5" LE "A11U"
    TOL-SPEC "0.001"

CONSTRAINT CD1A41L
F     COMMON / USRCA1 / A11U,A11L,A31UR,A31LT,A41U1,A41L1
    DEFINE D1A41 STREAM-PROP STREAM=HV-ISO PROPERTY=TBP-10
    SPEC "D1A41" GE "A41L1"
    TOL-SPEC "0.1"

CONSTRAINT CD1A41U
F     COMMON / USRCA1 / A11U,A11L,A31UR,A31LT,A41U1,A41L1
    DEFINE D1A41 STREAM-PROP STREAM=HV-ISO PROPERTY=TBP-10
    SPEC "D1A41" LE "A41U1"
    TOL-SPEC "0.1"

CONSTRAINT CD1A51L
F     COMMON / USRCA2 / A41U9,A41L9,A51U1,A51L1,A51U9,A51L9
    DEFINE D1A51 STREAM-PROP STREAM=LT-DIST PROPERTY=TBP-10
    SPEC "D1A51" GE "A51L1"
    TOL-SPEC "0.1"

CONSTRAINT CD1A51U
F     COMMON / USRCA2 / A41U9,A41L9,A51U1,A51L1,A51U9,A51L9
    DEFINE D1A51 STREAM-PROP STREAM=LT-DIST PROPERTY=TBP-10
    SPEC "D1A51" LE "A51U1"
    TOL-SPEC "0.1"

CONSTRAINT CD9A41L
F     COMMON / USRCA2 / A41U9,A41L9,A51U1,A51L1,A51U9,A51L9
    DEFINE D9A41 STREAM-PROP STREAM=HV-ISO PROPERTY=TBP-90
    SPEC "D9A41" GE "A41L9"
    TOL-SPEC "0.1"

CONSTRAINT CD9A41U
F     COMMON / USRCA2 / A41U9,A41L9,A51U1,A51L1,A51U9,A51L9
    DEFINE D9A41 STREAM-PROP STREAM=HV-ISO PROPERTY=TBP-90
```

```
    SPEC "D9A41" LE "A41U9"
    TOL-SPEC "0.1"

CONSTRAINT CD9A51L
F     COMMON / USRCA2 / A41U9,A41L9,A51U1,A51L1,A51U9,A51L9
    DEFINE D9A51 STREAM-PROP STREAM=LT-DIST PROPERTY=TBP-90
    SPEC "D9A51" GE "A51L9"
    TOL-SPEC "0.1"

CONSTRAINT CD9A51U
F     COMMON / USRCA2 / A41U9,A41L9,A51U1,A51L1,A51U9,A51L9
    DEFINE D9A51 STREAM-PROP STREAM=LT-DIST PROPERTY=TBP-90
    SPEC "D9A51" LE "A51U9"
    TOL-SPEC "0.1"

CONSTRAINT CF12L
F     COMMON / USRCF / F12U,F12L,F32U,F41U,F51U
    DEFINE F12 STREAM-VAR STREAM=C1RFLX SUBSTREAM=MIXED  &
        VARIABLE=STDVOL-FLOW
    SPEC "F12" GE "F12L"
    TOL-SPEC "1.0"

CONSTRAINT CF12U
F     COMMON / USRCF / F12U,F12L,F32U,F41U,F51U
    DEFINE F12 STREAM-VAR STREAM=C1RFLX SUBSTREAM=MIXED  &
        VARIABLE=STDVOL-FLOW
    SPEC "F12" LE "F12U"
    TOL-SPEC "1.0"

CONSTRAINT CF32U
F     COMMON / USRCF / F12U,F12L,F32U,F41U,F51U
    DEFINE F32 STREAM-VAR STREAM=C2RFLX SUBSTREAM=MIXED  &
        VARIABLE=STDVOL-FLOW
    SPEC "F32" LE "F32U"
    TOL-SPEC "1.0"

CONSTRAINT CF41U
F     COMMON / USRCF / F12U,F12L,F32U,F41U,F51U
    DEFINE F41 STREAM-VAR STREAM=IC1 SUBSTREAM=MIXED  &
        VARIABLE=STDVOL-FLOW
    SPEC "F41" LE "F41U"
    TOL-SPEC "1.0"

CONSTRAINT CF51U
F     COMMON / USRCF / F12U,F12L,F32U,F41U,F51U
```

```
    DEFINE F51 STREAM-VAR STREAM=IC2 SUBSTREAM=MIXED  &
        VARIABLE=STDVOL-FLOW
    SPEC "F51" LE "F51U"
    TOL-SPEC "1.0"

CONSTRAINT CRVPA31U
F     COMMON / USRCA1 / A11U,A11L,A31UR,A31LT,A41U1,A41L1
    DEFINE RVA31 STREAM-PROP STREAM=LT-ISO PROPERTY=RVP
    SPEC "RVA31" LE "A31UR"
    TOL-SPEC "0.01"

CONSTRAINT CT11L
F     COMMON / USRCT / T11U,T11L,T22U,T22L,T31U,T31L,T62U,T62L
    DEFINE T11 BLOCK-VAR BLOCK=C1 VARIABLE=TEMP  &
        SENTENCE=PROFILE ID1=3
    SPEC "T11" GE "T11L"
    TOL-SPEC "0.1"

CONSTRAINT CT11U
F     COMMON / USRCT / T11U,T11L,T22U,T22L,T31U,T31L,T62U,T62L
    DEFINE T11 BLOCK-VAR BLOCK=C1 VARIABLE=TEMP  &
        SENTENCE=PROFILE ID1=3
    SPEC "T11" LE "T11U"
    TOL-SPEC "0.1"

CONSTRAINT CT22L
F     COMMON / USRCT / T11U,T11L,T22U,T22L,T31U,T31L,T62U,T62L
    DEFINE T22 STREAM-VAR STREAM=C1REBL SUBSTREAM=MIXED  &
        VARIABLE=TEMP
    SPEC "T22" GE "T22L"
    TOL-SPEC "0.1"

CONSTRAINT CT22U
F     COMMON / USRCT / T11U,T11L,T22U,T22L,T31U,T31L,T62U,T62L
    DEFINE T22 STREAM-VAR STREAM=C1REBL SUBSTREAM=MIXED  &
        VARIABLE=TEMP
    SPEC "T22" LE "T22U"
    TOL-SPEC "0.1"

CONSTRAINT CT31L
F     COMMON / USRCT / T11U,T11L,T22U,T22L,T31U,T31L,T62U,T62L
    DEFINE T31 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=TEMP  &
        SENTENCE=PROFILE ID1=2
    SPEC "T31" GE "T31L"
    TOL-SPEC "0.1"
```

```
CONSTRAINT CT31U
F     COMMON / USRCT / T11U,T11L,T22U,T22L,T31U,T31L,T62U,T62L
    DEFINE T31 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=TEMP  &
        SENTENCE=PROFILE ID1=2
    SPEC "T31" LE "T31U"
    TOL-SPEC "0.1"

CONSTRAINT CT62L
F     COMMON / USRCT / T11U,T11L,T22U,T22L,T31U,T31L,T62U,T62L
    DEFINE T62 STREAM-VAR STREAM=C2REBL SUBSTREAM=MIXED  &
        VARIABLE=TEMP
    SPEC "T62" GE "T62L"
    TOL-SPEC "0.1"

CONSTRAINT CT62U
F     COMMON / USRCT / T11U,T11L,T22U,T22L,T31U,T31L,T62U,T62L
    DEFINE T62 STREAM-VAR STREAM=C2REBL SUBSTREAM=MIXED  &
        VARIABLE=TEMP
    SPEC "T62" LE "T62U"
    TOL-SPEC "0.1"

CONSTRAINT CTBUB31L
F     COMMON / USRCA1 / A11U,A11L,A31UR,A31LT,A41U1,A41L1
    DEFINE TBUB31 STREAM-PROP STREAM=LT-ISO PROPERTY=TBUB
    SPEC "TBUB31" GE "A31LT"
    TOL-SPEC "0.1"

CONSTRAINT EQC1Q1
F     COMMON / USRCEQ / EQ1,EQ2
    DEFINE C1Q1 BLOCK-VAR BLOCK=C1 VARIABLE=COND-DUTY  &
        SENTENCE=RESULTS
    SPEC "C1Q1" EQ "EQ1"
    TOL-SPEC "0.1"

CONSTRAINT EQC2Q1
F     COMMON / USRCEQ / EQ1,EQ2
    DEFINE C2Q1 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=COND-DUTY  &
        SENTENCE=RESULTS
    SPEC "C2Q1" EQ "EQ2"
    TOL-SPEC "0.1"

OPTIMIZATION DREC
F     COMMON / USRSF1 / SF01,SF11,SF12,SF13,SF21,SF22
F     COMMON / USRSF2 / SF31,SF32,SF41,SF42,SF51,SF52,SF61,SF62
```

```
F     COMMON / USRST1 / ST01,ST12,ST13,ST21,ST22,ST23
F     COMMON / USRST2 / ST31,ST32,ST41,ST42,ST51,ST52,ST61,ST62
F     COMMON / USRVF1 / VF01,VF11,VF12,VF13,VF21,VF22
F     COMMON / USRVF2 / VF31,VF32,VF41,VF42,VF51,VF52,VF61,VF62
F     COMMON / USRVT1 / VT01,VT12,VT13,VT21,VT22,VT23
F     COMMON / USRVT2 / VT31,VT32,VT41,VT42,VT51,VT52,VT61,VT62
F     COMMON / USREF1 / EF01,EF11,EF12,EF13,EF21,EF22
F     COMMON / USREF2 / EF31,EF32,EF41,EF42,EF51,EF52,EF61,EF62
F     COMMON / USRET1 / ET01,ET12,ET13,ET21,ET22,ET23
F     COMMON / USRET2 / ET31,ET32,ET41,ET42,ET51,ET52,ET61,ET62
F     COMMON / USRWF1 / WF01,WF11,WF12,WF13,WF21,WF22
F     COMMON / USRWF2 / WF31,WF32,WF41,WF42,WF51,WF52,WF61,WF62
F     COMMON / USRWT1 / WT01,WT12,WT13,WT21,WT22,WT23
F     COMMON / USRWT2 / WT31,WT32,WT41,WT42,WT51,WT52,WT61,WT62
F     COMMON / USRB1 / BSTABF,BSTABT,BC1RR,BC1RDV,BC1RBF,BC1RBT
F     COMMON / USRB2 / BE1T,BC2MD,BC2RBF,BC2RBT,BC2SCT,BC4QN
F     COMMON / USRB3 / BC5QN,BC2C4F,BC2C5F
    DEFINE F01 STREAM-VAR STREAM=STAB SUBSTREAM=MIXED  &
        VARIABLE=STDVOL-FLOW
    DEFINE F11 STREAM-PROP STREAM=VAP PROPERTY=V-STDVOL
    DEFINE F12 STREAM-VAR STREAM=C1RFLX SUBSTREAM=MIXED  &
        VARIABLE=STDVOL-FLOW
    DEFINE F13 STREAM-VAR STREAM=DIST-C4 SUBSTREAM=MIXED  &
        VARIABLE=STDVOL-FLOW
    DEFINE F21 STREAM-VAR STREAM=C1REBL SUBSTREAM=MIXED  &
        VARIABLE=STDVOL-FLOW
    DEFINE F22 STREAM-VAR STREAM=SPLTFEED SUBSTREAM=MIXED  &
        VARIABLE=STDVOL-FLOW
    DEFINE F31 STREAM-VAR STREAM=LT-ISO SUBSTREAM=MIXED  &
        VARIABLE=STDVOL-FLOW
    DEFINE F32 STREAM-VAR STREAM=C2RFLX SUBSTREAM=MIXED  &
        VARIABLE=STDVOL-FLOW
    DEFINE F41 STREAM-VAR STREAM=IC1 SUBSTREAM=MIXED  &
        VARIABLE=STDVOL-FLOW
    DEFINE F42 STREAM-VAR STREAM=HV-ISO SUBSTREAM=MIXED  &
        VARIABLE=STDVOL-FLOW
    DEFINE F51 STREAM-VAR STREAM=IC2 SUBSTREAM=MIXED  &
        VARIABLE=STDVOL-FLOW
    DEFINE F52 STREAM-VAR STREAM=LT-DIST SUBSTREAM=MIXED  &
        VARIABLE=STDVOL-FLOW
    DEFINE F61 STREAM-VAR STREAM=C2REBL SUBSTREAM=MIXED  &
        VARIABLE=STDVOL-FLOW
    DEFINE F62 STREAM-VAR STREAM=SSFEED SUBSTREAM=MIXED  &
        VARIABLE=STDVOL-FLOW
    DEFINE T01 STREAM-VAR STREAM=STAB SUBSTREAM=MIXED  &
```

```
    VARIABLE=TEMP
DEFINE T12 BLOCK-VAR BLOCK=C1 VARIABLE=TEMP  &
    SENTENCE=PROFILE ID1=2
DEFINE T13 STREAM-VAR STREAM=DIST-C4 SUBSTREAM=MIXED  &
    VARIABLE=TEMP
DEFINE T21 STREAM-VAR STREAM=SPLTFEED SUBSTREAM=MIXED  &
    VARIABLE=TEMP
DEFINE T22 STREAM-VAR STREAM=C1REBL SUBSTREAM=MIXED  &
    VARIABLE=TEMP
DEFINE T23 STREAM-VAR STREAM=SPLT SUBSTREAM=MIXED  &
    VARIABLE=TEMP
DEFINE T31 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=TEMP  &
    SENTENCE=PROFILE ID1=2
DEFINE T32 STREAM-VAR STREAM=LT-ISO SUBSTREAM=MIXED  &
    VARIABLE=TEMP
DEFINE T41 STREAM-VAR STREAM=IC1 SUBSTREAM=MIXED  &
    VARIABLE=TEMP
DEFINE T42 STREAM-VAR STREAM=HV-ISO SUBSTREAM=MIXED  &
    VARIABLE=TEMP
DEFINE T51 STREAM-VAR STREAM=IC2 SUBSTREAM=MIXED  &
    VARIABLE=TEMP
DEFINE T52 STREAM-VAR STREAM=LT-DIST SUBSTREAM=MIXED  &
    VARIABLE=TEMP
DEFINE T61 STREAM-VAR STREAM=SSFEED SUBSTREAM=MIXED  &
    VARIABLE=TEMP
DEFINE T62 STREAM-VAR STREAM=C2REBL SUBSTREAM=MIXED  &
    VARIABLE=TEMP
DEFINE C1RR BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-RR  &
    SENTENCE=COL-SPECS
DEFINE C1RDV BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-RDV  &
    SENTENCE=COL-SPECS
DEFINE C1RBFL BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=PUMPAROUND ID1=C1REBL
DEFINE C1RBT BLOCK-VAR BLOCK=C1 VARIABLE=TEMP  &
    SENTENCE=PUMPAROUND ID1=C1REBL
DEFINE E1T BLOCK-VAR BLOCK=E1 VARIABLE=TEMP SENTENCE=PARAM
DEFINE C2MD BLOCK-VAR BLOCK=C2C4C5 VARIABLE=MOLE-D  &
    SENTENCE=COL-SPECS
DEFINE C2RBFL BLOCK-VAR BLOCK=C2SPL1 SENTENCE=MOLE-FLOW  &
    VARIABLE=FLOW ID1=C2RBFD
DEFINE C2RBT BLOCK-VAR BLOCK=F2 VARIABLE=TEMP  &
    SENTENCE=PARAM
DEFINE C2SC BLOCK-VAR BLOCK=C2C4C5 VARIABLE=T1  &
    SENTENCE=COL-SPECS
DEFINE C4QN BLOCK-VAR BLOCK=C2C4C5 VARIABLE=Q-REB  &
```

```
      SENTENCE=STRIPPER ID1=C4
    DEFINE C5QN BLOCK-VAR BLOCK=C2C4C5 VARIABLE=Q-REB  &
      SENTENCE=STRIPPER ID1=C5
    DEFINE C2IC1 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=MOLE-DRAW  &
      SENTENCE=STRIPPER ID1=C4
    DEFINE C2IC2 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=MOLE-DRAW  &
      SENTENCE=STRIPPER ID1=C5
    DEFINE STABF STREAM-VAR STREAM=STAB SUBSTREAM=MIXED  &
      VARIABLE=MOLE-FLOW
    DEFINE STABT STREAM-VAR STREAM=STAB SUBSTREAM=MIXED  &
      VARIABLE=TEMP
C     CALCULATE THE WEIGHTED LEAST SQUARES ERROR TERMS
F     EF01 = WF01*((F01-SF01)/VF01)**2
F     EF11 = WF11*((F11-SF11)/VF11)**2
F     EF12 = WF12*((F12-SF12)/VF12)**2
F     EF13 = WF13*((F13-SF13)/VF13)**2
F     EF21 = WF21*((F21-SF21)/VF21)**2
F     EF22 = WF22*((F22-SF22)/VF22)**2
F     EF31 = WF31*((F31-SF31)/VF31)**2
F     EF32 = WF32*((F32-SF32)/VF32)**2
F     EF41 = WF41*((F41-SF41)/VF41)**2
F     EF42 = WF42*((F42-SF42)/VF42)**2
F     EF51 = WF51*((F51-SF51)/VF51)**2
F     EF52 = WF52*((F52-SF52)/VF52)**2
F     EF61 = WF61*((F61-SF61)/VF61)**2
F     EF62 = WF62*((F62-SF62)/VF62)**2
F     ET01 = WT01*((T01-ST01)/VT01)**2
F     ET12 = WT12*((T12-ST12)/VT12)**2
F     ET13 = WT13*((T13-ST13)/VT13)**2
F     ET21 = WT21*((T21-ST21)/VT21)**2
F     ET22 = WT22*((T22-ST22)/VT22)**2
F     ET23 = WT23*((T23-ST23)/VT23)**2
F     ET31 = WT31*((T31-ST31)/VT31)**2
F     ET32 = WT32*((T32-ST32)/VT32)**2
F     ET41 = WT41*((T41-ST41)/VT41)**2
F     ET42 = WT42*((T42-ST42)/VT42)**2
F     ET51 = WT51*((T51-ST51)/VT51)**2
F     ET52 = WT52*((T52-ST52)/VT52)**2
F     ET61 = WT61*((T61-ST61)/VT61)**2
F     ET62 = WT62*((T62-ST62)/VT62)**2
C     CALCULATE THE OVERALL OBJECTIVE FUNCTION
F     DROBJ = EF01+EF11+EF12+EF13+EF21+EF22+EF31+EF32+EF41+EF42
F    *       +EF51+EF52+EF61+EF62+ET01+ET12+ET13+ET21+ET22+ET23
F    *       +ET31+ET32+ET41+ET42+ET51+ET52+ET61+ET62
    MINIMIZE "DROBJ"
```

```
VARY STREAM-VAR STREAM=STAB SUBSTREAM=MIXED  &
    VARIABLE=MOLE-FLOW LABEL="STAB" "STREAM" "MOLE" "FLOW"
LIMITS "STABF - BSTABF" "STABF + BSTABF"
VARY STREAM-VAR STREAM=STAB SUBSTREAM=MIXED VARIABLE=TEMP  &
    LABEL="STAB" "STREAM" "TEMP"
LIMITS "STABT - BSTABT" "STABT + BSTABT"
VARY BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-RR SENTENCE=COL-SPECS  &
    LABEL="C1" "REFLUX" "RATIO"
LIMITS "C1RR - BC1RR" "C1RR + BC1RR"
VARY BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-RDV SENTENCE=COL-SPECS  &
    LABEL="C1" "DISTILL" "MOLE" "VAP FRAC"
LIMITS "C1RDV - BC1RDV" "C1RDV + BC1RDV"
VARY BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=PUMPAROUND ID1=C1REBL LABEL="C1" "REBOILER" "MOLE"  &
    "FLOW"
LIMITS "C1RBFL - BC1RBF" "C1RBFL + BC1RBF"
VARY BLOCK-VAR BLOCK=C1 VARIABLE=TEMP SENTENCE=PUMPAROUND  &
    ID1=C1REBL LABEL="C1" "REBOILER" "OUTLET" "TEMP"
LIMITS "C1RBT - BC1RBT" "C1RBT + BC1RBT"
VARY BLOCK-VAR BLOCK=E1 VARIABLE=TEMP SENTENCE=PARAM  &
    LABEL="E1" "OUTLET" "TEMP"
LIMITS "E1T - BE1T" "E1T + BE1T"
VARY BLOCK-VAR BLOCK=C2C4C5 VARIABLE=MOLE-D  &
    SENTENCE=COL-SPECS LABEL="C2 TOTAL" "DISTILL" "MOLE" "FLOW"
LIMITS "C2MD - BC2MD" "C2MD + BC2MD"
VARY BLOCK-VAR BLOCK=C2SPL1 SENTENCE=MOLE-FLOW VARIABLE=FLOW  &
    ID1=C2RBFD LABEL="C2RBFD" "C2RBFD" "C2RBFD" "C2RBFD"
LIMITS "C2RBFL - BC2RBF" "C2RBFL + BC2RBF"
VARY BLOCK-VAR BLOCK=F2 VARIABLE=TEMP SENTENCE=PARAM  &
    LABEL="C2" "REBOILER" "OUTLET" "TEMP"
LIMITS "C2RBT - BC2RBT" "C2RBT + BC2RBT"
VARY BLOCK-VAR BLOCK=C2C4C5 VARIABLE=T1 SENTENCE=COL-SPECS  &
    LABEL="C2" "CONDENSE" "SUBCOOL" "TEMP"
LIMITS "C2SC - BC2SCT" "C2SC + BC2SCT"
VARY BLOCK-VAR BLOCK=C2C4C5 VARIABLE=Q-REB SENTENCE=STRIPPER  &
    ID1=C4 LABEL="C4" "REBOILER" "HEAT" "DUTY"
LIMITS "C4QN - BC4QN" "C4QN + BC4QN"
VARY BLOCK-VAR BLOCK=C2C4C5 VARIABLE=Q-REB SENTENCE=STRIPPER  &
    ID1=C5 LABEL="C5" "REBOILER" "HEAT" "DUTY"
LIMITS "C5QN - BC5QN" "C5QN + BC5QN"
VARY BLOCK-VAR BLOCK=C2C4C5 VARIABLE=MOLE-DRAW  &
    SENTENCE=STRIPPER ID1=C4 LABEL="C2 -> C4" "CONNECT" "STREAM"  &
    "MOLEFLOW"
LIMITS "C2IC1 - BC2C4F" "C2IC1 + BC2C4F"
VARY BLOCK-VAR BLOCK=C2C4C5 VARIABLE=MOLE-DRAW  &
```

```
        SENTENCE=STRIPPER ID1=C5 LABEL="C2 -> C5" "CONNECT" "STREAM"  &
        "MOLEFLOW"
    LIMITS "C2IC2 - BC2C5F" "C2IC2 + BC2C5F"

OPTIMIZATION OPT
F     COMMON / USRD1 / DVAP,DDIST,DLTISO,DHVISO,DLTDST,DSSFED
F     COMMON / USRD2 / DSTAB,DFO
F     COMMON / USRPRM / PRMRVP
F     COMMON / USRSPC / SPCRVP
F     COMMON / USRCNV / CNVB2M,CNVBTM,CNVBTU,CNVP2K
F     COMMON / USRFEF / F1EFF,F2EFF
F     COMMON / USRPVL / PVAP,PDIST,PLTISO,PHVISO,PLTDST,PSSFED,PVALS
F     COMMON / USRCST / CSTAB,COF1,COF2
F     COMMON / USRQLT / QLTISO
F     COMMON / USRP1 / PC1RR,PC1RDV,PC1RBF,PC1RBT,PE1T,PC2MD,PC2RBF
F     COMMON / USRP2 / PC2RBT,PC2SCT,PC4QN,PC5QN,PC2C4F,PC2C5F
    DEFINE F01 STREAM-VAR STREAM=STAB SUBSTREAM=MIXED  &
        VARIABLE=STDVOL-FLOW
    DEFINE F11STD STREAM-VAR STREAM=VAP SUBSTREAM=MIXED  &
        VARIABLE=STDVOL-FLOW
    DEFINE F13 STREAM-VAR STREAM=DIST-C4 SUBSTREAM=MIXED  &
        VARIABLE=STDVOL-FLOW
    DEFINE F31 STREAM-VAR STREAM=LT-ISO SUBSTREAM=MIXED  &
        VARIABLE=STDVOL-FLOW
    DEFINE F42 STREAM-VAR STREAM=HV-ISO SUBSTREAM=MIXED  &
        VARIABLE=STDVOL-FLOW
    DEFINE F52 STREAM-VAR STREAM=LT-DIST SUBSTREAM=MIXED  &
        VARIABLE=STDVOL-FLOW
    DEFINE F62 STREAM-VAR STREAM=SSFEED SUBSTREAM=MIXED  &
        VARIABLE=STDVOL-FLOW
    DEFINE RVA31 STREAM-PROP STREAM=LT-ISO PROPERTY=RVP
    DEFINE C1QN BLOCK-VAR BLOCK=C1 VARIABLE=DUTY  &
        SENTENCE=PA-RESULTS ID1=C1REBL
    DEFINE C2QN BLOCK-VAR BLOCK=F2 VARIABLE=QCALC  &
        SENTENCE=PARAM
    DEFINE C1RR BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-RR  &
        SENTENCE=COL-SPECS
    DEFINE C1RDV BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-RDV  &
        SENTENCE=COL-SPECS
    DEFINE C1RBFL BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
        SENTENCE=PUMPAROUND ID1=C1REBL
    DEFINE C1RBT BLOCK-VAR BLOCK=C1 VARIABLE=TEMP  &
        SENTENCE=PUMPAROUND ID1=C1REBL
    DEFINE E1T BLOCK-VAR BLOCK=E1 VARIABLE=TEMP SENTENCE=PARAM
    DEFINE C2MD BLOCK-VAR BLOCK=C2C4C5 VARIABLE=MOLE-D  &
```

```
        SENTENCE=COL-SPECS
    DEFINE C2RBFL BLOCK-VAR BLOCK=C2SPL1 SENTENCE=MOLE-FLOW  &
        VARIABLE=FLOW ID1=C2RBFD
    DEFINE C2RBT BLOCK-VAR BLOCK=F2 VARIABLE=TEMP  &
        SENTENCE=PARAM
    DEFINE C2SC BLOCK-VAR BLOCK=C2C4C5 VARIABLE=T1  &
        SENTENCE=COL-SPECS
    DEFINE C4QN BLOCK-VAR BLOCK=C2C4C5 VARIABLE=Q-REB  &
        SENTENCE=STRIPPER ID1=C4
    DEFINE C5QN BLOCK-VAR BLOCK=C2C4C5 VARIABLE=Q-REB  &
        SENTENCE=STRIPPER ID1=C5
    DEFINE C2IC1 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=MOLE-DRAW  &
        SENTENCE=STRIPPER ID1=C4
    DEFINE C2IC2 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=MOLE-DRAW  &
        SENTENCE=STRIPPER ID1=C5
C REAL TIME OPTIMIZATION OBJECTIVE FUNCTION
C   PROFIT = (PRODUCT VALUES) - (FEED COST) - (UTILITY COST)
C     BASIS: 1 HR
C   PREMIUMS: [($/M3)=($/M3)+($/M3/UNIT)*(UNIT-UNIT)]
F     QLTISO = DLTISO+(PRMRVP*((RVA31*CNVP2K)-SPCRVP))
C   PRODUCT VALUES: [($/HR)=($/M3)*(M3 DAY/HR BBL)*(BBL/DAY)]
F     PVAP = DVAP*CNVB2M*F11STD
F     PDIST = DDIST*CNVB2M*F13
F     PLTISO = QLTISO*CNVB2M*F31
F     PHVISO = DHVISO*CNVB2M*F42
F     PLTDST = DLTDST*CNVB2M*F52
F     PSSFED = DSSFED*CNVB2M*F62
F     PVALS = PVAP+PDIST+PLTISO+PHVISO+PLTDST+PSSFED
C   FEED COST: [($/HR)=($/M3)*(M3 DAY/HR BBL)*(BBL/DAY)]
F     CSTAB = DSTAB*CNVB2M*F01
C   UTILITY COSTS: [($/HR)=($/LFEM3)*(LFEM3/LFEBBL)*(LFEBBL/MMBTU)
C                          *(MMBTU/HR)*(1/FEFF)]
F     COF1 = DF0*CNVBTM*CNVBTU*C1QN*(1/F1EFF)
F     COF2 = DF0*CNVBTM*CNVBTU*C2QN*(1/F2EFF)
C   PROFIT:
F     PROFIT = PVALS-CSTAB-COF1-COF2
    MAXIMIZE "PROFIT"
    CONSTRAINTS C5A11L / C5A11U / CRVPA31U / CD1A41L /  &
        CD1A41U / CD1A51L / CD1A51U / CD9A41L / CD9A41U /  &
        CD9A51L / CD9A51U / CF12L / CF12U / CF32U / CF41U  &
        / CF51U / CT11L / CT11U / CT22L / CT22U / CT31L /  &
        CT31U / CT62L / CT62U / CTBUB31L / EQC1Q1 / EQC2Q1
    VARY BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-RR SENTENCE=COL-SPECS  &
        LABEL="C1" "MOLAR" "REFLUX" "RATIO"
    LIMITS "C1RR-(C1RR*PC1RR)" "C1RR+(C1RR*PC1RR)"
```

```
    VARY BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-RDV SENTENCE=COL-SPECS  &
        LABEL="C1" "DISTILL" "MOLE" "VAP FRAC"
    LIMITS "C1RDV-(C1RDV*PC1RDV)" "C1RDV+(C1RDV*PC1RDV)"
    VARY BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
        SENTENCE=PUMPAROUND ID1=C1REBL LABEL="C1" "REBOILER" "MOLE"  &
        "FLOW"
    LIMITS "C1RBFL-(C1RBFL*PC1RBF)" "C1RBFL+(C1RBFL*PC1RBF)"
    VARY BLOCK-VAR BLOCK=C1 VARIABLE=TEMP SENTENCE=PUMPAROUND  &
        ID1=C1REBL LABEL="C1" "REBOILER" "OUTLET" "TEMP"
    LIMITS "C1RBT - PC1RBT" "C1RBT + PC1RBT"
    VARY BLOCK-VAR BLOCK=E1 VARIABLE=TEMP SENTENCE=PARAM  &
        LABEL="E1" "OUTLET" "TEMP"
    LIMITS "E1T - PE1T" "E1T + PE1T"
    VARY BLOCK-VAR BLOCK=C2C4C5 VARIABLE=MOLE-D  &
        SENTENCE=COL-SPECS LABEL="C2 TOTAL" "DIST" "MOLE" "FLOW"
    LIMITS "C2MD-(C2MD*PC2MD)" "C2MD+(C2MD*PC2MD)"
    VARY BLOCK-VAR BLOCK=C2SPL1 SENTENCE=MOLE-FLOW VARIABLE=FLOW  &
        ID1=C2RBFD LABEL="C2RBFD" "C2RBFD" "C2RBFD" "C2RBFD"
    LIMITS "C2RBFL-(C2RBFL*PC2RBF)" "C2RBFL+(C2RBFL*PC2RBF)"
    VARY BLOCK-VAR BLOCK=F2 VARIABLE=TEMP SENTENCE=PARAM  &
        LABEL="C2" "REBOILER" "OUTLET" "TEMP"
    LIMITS "C2RBT - PC2RBT" "C2RBT + PC2RBT"
    VARY BLOCK-VAR BLOCK=C2C4C5 VARIABLE=T1 SENTENCE=COL-SPECS  &
        LABEL="C2" "CONDENSE" "SUBCOOL" "TEMP"
    LIMITS "C2SC - PC2SCT" "C2SC + PC2SCT"
    VARY BLOCK-VAR BLOCK=C2C4C5 VARIABLE=Q-REB SENTENCE=STRIPPER  &
        ID1=C4 LABEL="C4" "REBOILER" "HEAT" "DUTY"
    LIMITS "C4QN-(C4QN*PC4QN)" "C4QN+(C4QN*PC4QN)"
    VARY BLOCK-VAR BLOCK=C2C4C5 VARIABLE=Q-REB SENTENCE=STRIPPER  &
        ID1=C5 LABEL="C5" "REBOILER" "HEAT" "DUTY"
    LIMITS "C5QN-(C5QN*PC5QN)" "C5QN+(C5QN*PC5QN)"
    VARY BLOCK-VAR BLOCK=C2C4C5 VARIABLE=MOLE-DRAW  &
        SENTENCE=STRIPPER ID1=C4 LABEL="C2 -> C4" "CONNECT" "STREAM"  &
        "MOLEFLOW"
    LIMITS "C2IC1-(C2IC1*PC2C4F)" "C2IC1+(C2IC1*PC2C4F)"
    VARY BLOCK-VAR BLOCK=C2C4C5 VARIABLE=MOLE-DRAW  &
        SENTENCE=STRIPPER ID1=C5 LABEL="C2 -> C5" "CONNECT" "STREAM"  &
        "MOLEFLOW"
    LIMITS "C2IC2-(C2IC2*PC2C5F)" "C2IC2+(C2IC2*PC2C5F)"

CONV-OPTIONS
    PARAM TRACEOPT=GRADUAL CHECKSEQ=YES

CONVERGENCE C2REBL WEGSTEIN
    DESCRIPTION "C2 REBOILER TEAR STREAM"
```

```
    BLOCK-OPTION CONV-LEVEL=3 TERM-LEVEL=2 RESTART=YES
    TEAR C2REBL

CONVERGENCE DREC SQP
    DESCRIPTION "SQP FOR DATA RECONCILIATION"
    OPTIMIZE OPTID=DREC
    PARAM MAXIT=300 MAXPASS=99998 TOL=.10 MAXLSPASS=4  &
        EST-STEP=YES CONST-ITER=10 DERIV-SWITCH=YES

CONVERGENCE OPT SQP
    DESCRIPTION "SQP FOR OPTIMIZATION"
    OPTIMIZE OPTID=OPT
    PARAM MAXIT=300 MAXPASS=99998 TOL=.10 MAXLSPASS=4  &
        EST-STEP=YES CONST-ITER=10 DERIV-SWITCH=YES

SEQUENCE RTO DRSPCSET DREC C1 E1 C2REBL C2C4C5 C2SPL1 F2  &
        (RETURN C2REBL) (RETURN DREC) DRDATA OPSPCSET OPT C1  &
        E1 C2REBL C2C4C5 C2SPL1 F2 (RETURN C2REBL) (RETURN OPT)  &
        OPDATA

REPORT NOINSERT NOADA NOSENSITIVIT NOPROPERTIES NOCOSTBLOCK  &
        NOUNITS NOUTILITIES NOECONOMIC

BLOCK-REPORT NOSORT NOTOTBAL NOINPUT

STREAM-REPOR WIDE NOMOLEFLOW MOLEFRAC NOVOLFRAC NOATTR-DESC  &
        NOCOMP-ATTR NOSUBS-ATTR PROPERTIES=TBP-10 TBP-90 STDVOL  &
        V-STDVOL VOL RVP TBUB EXCL-STREAMS=C2BOT C2RBFD

FLOWSHEET-RE NODESCRIPTIO NOTOTBAL NOCOMPBAL NOSEQUENCE  &
        NOFORTRAN NODESIGN-SPE NOCONSTRAINT NOOPTIMIZATI  &
        NOTRANSFER
```

Appendix E

SPEEDUP Real Time Optimizer Code

E.1 setup

The following executable C-shell script, setup, is used to setup the SPEED-UP real time optimizer.

```
#!/bin/csh
#
# SPEEDUP RTO SETUP SCRIPT
#
# Sets up the SpeedUp RTO simulation and plant simulations in
# preparation for real time optimzition runs.
#
# Set up plant
echo "=== SETTING UP SPEEDUP REAL TIME OPTIMIZER SIMULATIONS ==="
cd plant
if ( !(-e plant6.inp) ) then
  echo "=== SETUP: ERROR: plant/plant6.inp does not exist ==="
  exit
endif
if ( !(-e randdata.f) ) then
  echo "=== SETUP: ERROR: plant/randdata.f does not exist ==="
  exit
endif
echo "=== SETUP: Setting up ASPEN PLUS plant simulation ==="
```

```
aspen plant6 /itonly
# Set up randdata.exe
echo "=== SETUP: Setting up variable noise FORTRAN ==="
f77 randdata.f -l nag -o randdata.exe
# Set up Speedup Properties
cd ..
cd props
if ( !(-e prop6.inp) ) then
  echo "=== SETUP: ERROR: props/prop6.inp does not exist ==="
  exit
endif
echo "=== SETUP: Setting up SPEEDUP properties ==="
sairun prop6 z
setenv LIBPHPR 'pwd'
# Set up Speedup Data Rec
cd ..
cd rtodr
if ( !(-e model6x.speedup) ) then
  echo "=== SETUP: ERROR: rtodr/model6x.speedup does not exist ==="
  exit
endif
if ( !(-e report6y.speedup) ) then
  echo "=== SETUP: ERROR: rtodr/report6y.speedup does not exist ==="
  exit
endif
if ( !(-e rtodr6.f) ) then
  echo "=== SETUP: ERROR: rtodr/rtodr6.f does not exist ==="
  exit
endif
if ( !(-e rtodr6.speedup) ) then
  echo "=== SETUP: ERROR: rtodr/rtodr6.speedup does not exist ==="
  exit
endif
if ( !(-e rtodr6run.cmd) ) then
  echo "=== SETUP: ERROR: rtodr/rtodr6run.cmd does not exist ==="
  exit
endif
if ( !(-e rtodr6setup.cmd) ) then
  echo "=== SETUP: ERROR: rtodr/rtodr6setup.cmd does not exist ==="
  exit
endif
if ( !(-e speedup.include) ) then
  echo "=== SETUP: ERROR: rtodr/speedup.include does not exist ==="
  exit
endif
```

```
echo "=== SETUP: Setting up SPEEDUP RTO data rec. simulation ==="
fortsp rtodr6.f
speedup < rtodr6setup.cmd
# Set up Speedup Optimization
cd ..
cd rtopt
if ( !(-e model6x.speedup) ) then
  echo "=== SETUP: ERROR: rtopt/model6x.speedup does not exist ==="
  exit
endif
if ( !(-e report6y.speedup) ) then
  echo "=== SETUP: ERROR: rtopt/report6y.speedup does not exist ==="
  exit
endif
if ( !(-e rtopt6.f) ) then
  echo "=== SETUP: ERROR: rtopt/rtopt6.f does not exist ==="
  exit
endif
if ( !(-e rtopt6.speedup) ) then
  echo "=== SETUP: ERROR: rtopt/rtopt6.speedup does not exist ==="
  exit
endif
if ( !(-e rtopt6run.cmd) ) then
  echo "=== SETUP: ERROR: rtopt/rtopt6run.cmd does not exist ==="
  exit
endif
if ( !(-e rtopt6setup.cmd) ) then
  echo "=== SETUP: ERROR: rtopt/rtopt6setup.cmd does not exist ==="
  exit
endif
if ( !(-e speedup.include) ) then
  echo "=== SETUP: ERROR: rtopt/speedup.include does not exist ==="
  exit
endif
echo "=== SETUP: Setting up SPEEDUP RTO optimization simulation ==="
fortsp rtopt6.f
speedup < rtopt6setup.cmd
echo "=== SETUP: COMPLETED ==="
```

E.2 run

The following executable C-shell script,run, is used to run a given number of cycles of the SPEEDUP real time optimizer.

```
#!/bin/csh
#
# SPEEDUP RTO RUN SCRIPT
#
# Runs the SpeedUp RTO simulation and plant simulations.
#
# SYNTAX: run n
#   where n = number of cycles of the real time optimzer to be run
#             (default = 1 if n not given)
#
echo "=== SPEEDUP REAL TIME OPTIMIZER SIMULATION ==="
# Check number of cycles to be performed
if ( "$1" >= 1 ) then
  set MAXITN = $1
else
  set MAXITN = 1
endif
set ITN = 1
# Set Properties Environment Variable
cd props
setenv LIBPHPR `pwd`
cd ..
# Make results directory, deleting old one first
if ( -e results ) then
  rm -r -f results
endif
mkdir results
cd results
mkdir start
cd ..
while ( $MAXITN >= $ITN )
  echo "=== REAL TIME OPTIMIZER CYCLE $ITN BEGINNING ==="
  cd results
  set RESITN = "itn$ITN"
  mkdir $RESITN
  cd ..
#
# SPEEDUP RTODR6 RUN
#
  cd rtodr
# Check if input files all exist
  if ( !(-e IMEAS.DAT) ) then
    echo "=== RUN: ERROR: rtodr/IMEAS.DAT does not exist ==="
    exit
  endif
```

```
  if ( !(-e IVARNC.DAT) ) then
    echo "=== RUN: ERROR: rtodr/IVARNC.DAT does not exist ==="
    exit
  endif
  if ( !(-e IWGHT.DAT) ) then
    echo "=== RUN: ERROR: rtodr/IWGHT.DAT does not exist ==="
    exit
  endif
  if ( !(-e IDFREE.DAT) ) then
    echo "=== RUN: ERROR: rtodr/IDFREE.DAT does not exist ==="
    exit
  endif
  if ( !(-e ISET.DAT) ) then
    echo "=== RUN: ERROR: rtodr/ISET.DAT does not exist ==="
    exit
  endif
  if ( !(-e drpreset.speedup) ) then
    echo "=== RUN: ERROR: rtodr/drpreset.speedup does not exist ==="
    exit
  endif
# Remove any output files if they exist
  if ( -e RDSET.DAT ) then
    rm RDSET.DAT
  endif
  if ( -e oppreset.speedup ) then
    rm oppreset.speedup
  endif
  if ( -e speedup.lis ) then
    rm speedup.lis
  endif
# Run SpeedUp rtodr6 job
  echo "=== RUN: Starting SPEEDUP RTO Data Reconciliation Simulation ==="
  speedup < rtodr6run.cmd
# Check if simulation ran to completion
  if ( !(-e RDSET.DAT) ) then
    echo "=== RUN: ERROR: an error occured running speedup rtodr6 ==="
    exit
  endif
# Copy run results to results directory
  cp RDSET.DAT ../results/$RESITN
  cp ISET.DAT ../results/$RESITN
  if ("$RESITN" == "itn1") then
    cp drpreset.speedup ../results/start
    cp IMEAS.DAT ../results/start
    cp ISET.DAT ../results/start
```

```
  endif
  cp speedup.lis ../results/$RESITN/rtodr6.lis
# Send files to Optimzation run
  cp RDSET.DAT ../rtopt/ISET.DAT
  cp oppreset.speedup ../rtopt
  cd ..
#
# SPEEDUP RTOPT6 RUN
#
  cd rtopt
# Check if input files all exist
  if ( !(-e IOBJFN.DAT) ) then
    echo "=== RUN: ERROR: rtopt/IOBJFN.DAT does not exist ==="
    exit
  endif
  if ( !(-e ICNSTR.DAT) ) then
    echo "=== RUN: ERROR: rtopt/ICNSTR.DAT does not exist ==="
    exit
  endif
  if ( !(-e IOFREE.DAT) ) then
    echo "=== RUN: ERROR: rtopt/IOFREE.DAT does not exist ==="
    exit
  endif
  if ( !(-e ISET.DAT) ) then
    echo "=== RUN: ERROR: rtopt/ISET.DAT does not exist ==="
    exit
  endif
  if ( !(-e oppreset.speedup) ) then
    echo "=== RUN: ERROR: rtopt/oppreset.speedup does not exist ==="
    exit
  endif
# Remove any output files if they exist
  if ( -e ROSMST.DAT ) then
    rm ROSMST.DAT
  endif
  if ( -e ROSET.DAT ) then
    rm ROSET.DAT
  endif
  if ( -e drpreset.speedup ) then
    rm drpreset.speedup
  endif
  if ( -e speedup.lis ) then
    rm speedup.lis
  endif
# Run SpeedUp rtopt6 job
```

```
  echo "=== RUN: Starting SPEEDUP RTO Optimization Simulation ==="
  speedup < rtopt6run.cmd
# Check if simulation ran to completion
  if ( !(-e ROSET.DAT) ) then
    echo "=== RUN: ERROR: an error occured running speedup rtopt6 ==="
    exit
  endif
# Copy run results to results directory
  cp R*.DAT ../results/$RESITN
  cp speedup.lis ../results/$RESITN/rtopt6.lis
# Send files to Data Rec run, next cycle
  cp ROSET.DAT ../rtodr/ISET.DAT
  cp drpreset.speedup ../rtodr
#
# ASPEN PLANT6 RUN
#
# Set up for plant6 run
  cp ROSMST.DAT ../plant
  cd ..
  cd plant
# Check if input files all exist
  if ( !(-e ROSMST.DAT) ) then
    echo "=== RUN: ERROR: plant/ROSMST.DAT does not exist ==="
    exit
  endif
# Remove any output files if they exist
  if ( -e PLVARS.DAT ) then
    rm PLVARS.DAT
  endif
# Run Aspen Plus plant6 job
  echo "=== RUN: Starting ASPEN PLUS PLANT simulation ==="
  aspen plant6 /sponly
# Check if simulation ran to completion
  if ( !(-e PLVARS.DAT) ) then
    echo "=== RUN: ERROR: an error occured running aspen plant6 ==="
    exit
  endif
# Copy run results to results directory
  cp PLVARS.DAT ../results/$RESITN
#
# FORTRAN RANDDATA RUN
#
# Check if input files all exist
  if ( !(-e PLVARS.DAT) ) then
    echo "=== RUN: ERROR: plant/PLVARS.DAT does not exist ==="
```

```
    exit
  endif
  if ( !(-e RVARNC.DAT) ) then
    echo "=== RUN: ERROR: plant/RVARNC.DAT does not exist ==="
    exit
  endif
# Remove any output files if they exist
  if ( -e IMEAS.DAT ) then
    rm IMEAS.DAT
  endif
# Run FORTRAN randdata.exe job
  echo "=== RUN: Starting randdata.exe ==="
  randdata.exe
# Check if simulation ran to completion
  if ( !(-e IMEAS.DAT) ) then
    echo "=== RUN: ERROR: an error occured running randdata.exe ==="
    exit
  endif
# Copy run results to results directory
  cp IMEAS.DAT ../results/$RESITN
# Send files to Data Rec run, next cycle
  cp IMEAS.DAT ../rtodr
  cd ..
  set ITN = `expr $ITN + 1`
end
cp plant/plant6.rep results
echo "=== $MAXITN REAL TIME OPTIMIZER CYCLES COMPLETED ==="
```

E.3 Common Code from rtodr6.speedup and rtopt6.speedup

The following sections of SPEEDUP input code appear in both `rtodr6.speedup` and `rtopt6.speedup`. See section E.4 for the `rtodr6.speedup` specific code, and section E.6 for the `rtopt6.speedup` specific code.

```
DECLARE
# Taken from the Library Declare and modified where necessary
  TYPE

#                    ----   US UNITS OF MEASUREMENT ----
# THE VARIABLE TYPES USED IN THIS SIMULATION:
```

```
#                      DEFAULT LOWER     UPPER    SCALING       UNIT OF
#                      VALUE   BOUND     BOUND    FACTOR        MEASUREMENT
    dens_mol_liq     = 2.5     : 0.01    : 7.5    : 1       UNIT= "lbmol/ft3"
    dens_mol_vap     = 5.0E-4  : 5.0E-7  : 5      : 0.01    UNIT= "lbmol/ft3"
    enthflow         = 1.0     :-350     : 200    : 300     UNIT= "MMBtu/h"
    enth_mol         = 0.050   :-1.0     : 1.0    : 1       UNIT= "MMBtu/lbmol"
    enth_mol_liq     = 0.025   :-1.0     : 1.0    : 1       UNIT= "MMBtu/lbmol"
    enth_mol_vap     = 0.060   :-1.0     : 1.0    : 1       UNIT= "MMBtu/lbmol"
    flow_mol         = 1000.0  : 0.0     : 10000  : 5000    UNIT= "lbmol/h"
    flow_mol_liq     = 1000.0  : 0.0     : 6000   : 3000    UNIT= "lbmol/h"
    flow_mol_vap     = 1000.0  : 0.0     : 6000   : 3000    UNIT= "lbmol/h"
    flow_vol         = 5000.0  : 0.0     : 5.0E5  : 20000   UNIT= "ft3/h"
    flow_vol_liq     = 4000.0  : 0.0     : 5.0E5  : 15000   UNIT= "ft3/h"
    flow_vol_vap     = 10000   : 0.0     : 1.0E6  : 40000   UNIT= "ft3/h"
    molefraction     = 0.05    : 0.0     : 1.0    : 1       UNIT= "lbmol/lbmol"
    pressure         = 14.7    : 1.0     : 1470.0 : 200     UNIT= "psia"
    press_drop       = 164     :-300     : 300.0  : 200     UNIT= "psi"
    ratio            = 1       : 0.0     : 100.0  : 10      UNIT= "-"
    temperature      = 100.0   : 20.0    : 800.0  : 1000    UNIT= "F"
    temp_drop        = 5       :-200.0   : 200.0  : 200     UNIT= "F"
    temp_rise        = -47     :-200.0   : 200.0  : 200     UNIT= "F"
    vapfraction      = 0.10    : 0.0     : 1.0    : 1       UNIT= "lbmol/lbmol"
# Default stream type MAINSTREAM
  STREAM MAINSTREAM
    SET
      NOCOMP=23
    TYPE
      FLOW_MOL, MOLEFRACTION(NOCOMP), TEMPERATURE, PRESSURE, ENTH_MOL
    COMPONENTS
      H2, METHANE, ETHANE, PROPANE, IBUTANE, BUTANE,
      IPENTANE, PENTANE, B45T65, B65T85, B85T105, B105T125,
      B125T145, B145T165, B165T185, B185T205, B205T235, B235T265,
      B265T295, B295T325, B325T355, B355T395, B395T510
    OPTIONS
      OPSET = "PENG-ROB"
  STREAM LIQUID
    SET
      NOCOMP=23
    TYPE
      FLOW_MOL_LIQ, MOLEFRACTION(NOCOMP), TEMPERATURE, PRESSURE, ENTH_MOL_LIQ
    THERMO MAINSTREAM
  STREAM VAPOUR
    SET
      NOCOMP=23
    TYPE
```

```
      FLOW_MOL_VAP, MOLEFRACTION(NOCOMP), TEMPERATURE, PRESSURE, ENTH_MOL_VAP
    THERMO MAINSTREAM
  STREAM RVPMAIN
    SET
      NOCOMP = 24
    TYPE
      FLOW_MOL, MOLEFRACTION(NOCOMP), TEMPERATURE, PRESSURE, ENTH_MOL
    COMPONENTS
      H2, METHANE, ETHANE, PROPANE, IBUTANE, BUTANE,
      IPENTANE, PENTANE, B45T65, B65T85, B85T105, B105T125,
      B125T145, B145T165, B165T185, B185T205, B205T235, B235T265,
      B265T295, B295T325, B325T355, B355T395, B395T510, AIR
    OPTIONS
      OPSET = "SYSOP0"
  STREAM RVPLIQ
    SET
      NOCOMP=24
    TYPE
      FLOW_MOL_LIQ, MOLEFRACTION(NOCOMP), TEMPERATURE, PRESSURE, ENTH_MOL_LIQ
    THERMO RVPMAIN
  STREAM RVPVAP
    SET
      NOCOMP=24
    TYPE
      FLOW_MOL_VAP, MOLEFRACTION(NOCOMP), TEMPERATURE, PRESSURE, ENTH_MOL_VAP
    THERMO RVPMAIN
****
FLOWSHEET
# Molar Feed to the Feed Tray of C1
  STREAM STAB      OUTPUT   OF MFEED    IS INPUT   OF FO1
                   OUTPUT   OF FO1      IS INPUT 3 OF C1FT14
# Connect C1 Feed Tray to Stripping Section
                   OUTPUT 1 OF C1FT14   IS INPUT 1 OF C1S      TYPE VAPOUR
                   OUTPUT 2 OF C1S      IS INPUT 2 OF C1FT14   TYPE LIQUID
# Connect C1 Stripping Section to Partial Condenser and Reflux Splitter
                   OUTPUT 1 OF C1S      IS INPUT   OF C1COND   TYPE VAPOUR
  STREAM VAP       OUTPUT 1 OF C1COND   IS INPUT   OF F11      TYPE VAPOUR
                   OUTPUT   OF F11      IS PRODUCT 1           TYPE VAPOUR
                   OUTPUT 2 OF C1COND   IS INPUT 1 OF C1SPL1   TYPE LIQUID
  STREAM DIST_C4   OUTPUT 1 OF C1SPL1   IS INPUT   OF F13      TYPE LIQUID
                   OUTPUT   OF F13      IS PRODUCT 2           TYPE LIQUID
  STREAM C1RFLX    OUTPUT 2 OF C1SPL1   IS INPUT   OF F12      TYPE LIQUID
                   OUTPUT   OF F12      IS INPUT 2 OF C1S      TYPE LIQUID
# Connect C1 Feed Tray to Rectifying Section
                   OUTPUT 2 OF C1FT14   IS INPUT 2 OF C1R      TYPE LIQUID
```

```
                  OUTPUT 1 OF C1R      IS INPUT 1 OF C1FT14   TYPE VAPOUR
# Connect C1 Rectifying Section to Reboiler
                  OUTPUT 2 OF C1R      IS INPUT 1 OF C1BT29   TYPE LIQUID
                  OUTPUT 1 OF C1BT29   IS INPUT 1 OF C1R      TYPE VAPOUR
  STREAM SPLTFEED OUTPUT 2 OF C1BT29   IS INPUT   OF F22A     TYPE LIQUID
                  OUTPUT   OF F22A     IS INPUT   OF E1       TYPE LIQUID
                  OUTPUT 3 OF C1BT29   IS INPUT   OF C1REBL   TYPE LIQUID
  STREAM C1REBL   OUTPUT   OF C1REBL   IS INPUT   OF F21
                  OUTPUT   OF F21      IS INPUT   OF C1TR1
                  OUTPUT   OF C1TR1    IS INPUT 2 OF C1BT29
# Connect SPLTFEED to E1 Heat Exchanger
   STREAM SPLT    OUTPUT   OF E1       IS INPUT   OF F22
                  OUTPUT   OF F22      IS INPUT 3 OF C2FT21
# Connect C2 Feed Tray to Rectifying Section
                  OUTPUT 2 OF C2FT21   IS INPUT 2 OF C2R      TYPE LIQUID
                  OUTPUT 1 OF C2R      IS INPUT 1 OF C2FT21   TYPE VAPOUR
# Connect C2 Rectifying Section to Reboiler
                  OUTPUT 2 OF C2R      IS INPUT 1 OF C2BT24   TYPE LIQUID
                  OUTPUT 1 OF C2BT24   IS INPUT 1 OF C2R      TYPE VAPOUR
  STREAM SSFEED   OUTPUT 2 OF C2BT24   IS INPUT   OF F62      TYPE LIQUID
                  OUTPUT   OF F62      IS PRODUCT 6           TYPE LIQUID
                  OUTPUT 3 OF C2BT24   IS INPUT   OF C2REBL   TYPE LIQUID
  STREAM C2REBL   OUTPUT   OF C2REBL   IS INPUT   OF F61
                  OUTPUT   OF F61      IS INPUT   OF C2TR3
                  OUTPUT   OF C2TR3    IS INPUT 2 OF C2BT24
# Connect C2 Feed Tray to Stripping Section
                  OUTPUT 1 OF C2FT21   IS INPUT 1 OF C2S3     TYPE VAPOUR
                  OUTPUT 2 OF C2S3     IS INPUT 2 OF C2FT21   TYPE LIQUID
# Build C2 Stripping Section
                  OUTPUT 1 OF C2S3     IS INPUT 1 OF C2T18    TYPE VAPOUR
                  OUTPUT 1 OF C2T18    IS INPUT 1 OF C2FT17   TYPE VAPOUR
                  OUTPUT 2 OF C2SPL2   IS INPUT 2 OF C2S3     TYPE LIQUID
  STREAM IC2      OUTPUT 1 OF C2SPL2   IS INPUT   OF F51      TYPE LIQUID
                  OUTPUT   OF F51      IS INPUT 2 OF C5       TYPE LIQUID
                  OUTPUT 2 OF C2FT17   IS INPUT 2 OF C2T18    TYPE LIQUID
                  OUTPUT 2 OF C2T18    IS INPUT 1 OF C2SPL2   TYPE LIQUID
  STREAM C5RTN1   OUTPUT 1 OF C5       IS INPUT   OF C2TR2    TYPE VAPOUR
  STREAM C5RTN2   OUTPUT   OF C2TR2    IS INPUT 3 OF C2FT17   TYPE VAPOUR
                  OUTPUT 1 OF C2FT17   IS INPUT 1 OF C2T16    TYPE VAPOUR
                  OUTPUT 2 OF C2T16    IS INPUT 2 OF C2FT17   TYPE LIQUID
                  OUTPUT 1 OF C2T16    IS INPUT 1 OF C2T15    TYPE VAPOUR
                  OUTPUT 2 OF C2SPL1   IS INPUT 2 OF C2T16    TYPE LIQUID
  STREAM IC1      OUTPUT 1 OF C2SPL1   IS INPUT   OF F41      TYPE LIQUID
                  OUTPUT   OF F41      IS INPUT 2 OF C4       TYPE LIQUID
                  OUTPUT 1 OF C2T15    IS INPUT 1 OF C2FT14   TYPE VAPOUR
```

```
                  OUTPUT 2 OF C2T15    IS INPUT 1 OF C2SPL1   TYPE LIQUID
                  OUTPUT 2 OF C2FT14   IS INPUT 2 OF C2T15    TYPE LIQUID
  STREAM C4RTN1   OUTPUT 1 OF C4       IS INPUT   OF C2TR1    TYPE VAPOUR
  STREAM C4RTN2   OUTPUT   OF C2TR1    IS INPUT 3 OF C2FT14   TYPE VAPOUR
                  OUTPUT 1 OF C2FT14   IS INPUT 1 OF C2S1     TYPE VAPOUR
                  OUTPUT 2 OF C2S1     IS INPUT 2 OF C2FT14   TYPE LIQUID
# Connect C2 Stripping Section to Total Condenser
                  OUTPUT 1 OF C2S1     IS INPUT   OF C2COND   TYPE VAPOUR
  STREAM LT_ISO   OUTPUT 2 OF C2COND   IS INPUT   OF F31      TYPE LIQUID
                  OUTPUT   OF F31      IS INPUT   OF M2R31    TYPE LIQUID
                  OUTPUT   OF M2R31    IS INPUT   OF RVP31    TYPE RVPLIQ
                  OUTPUT   OF RVP31    IS PRODUCT 3           TYPE RVPLIQ
  STREAM C2RFLX   OUTPUT 1 OF C2COND   IS INPUT   OF F32      TYPE LIQUID
                  OUTPUT   OF F32      IS INPUT 2 OF C2S1     TYPE LIQUID
# Connect C5 Reboiler
                  OUTPUT 2 OF C5       IS INPUT   OF C5REBL   TYPE LIQUID
                  OUTPUT 1 OF C5REBL   IS INPUT 1 OF C5       TYPE VAPOUR
  STREAM HV_ISO   OUTPUT 2 OF C5REBL   IS INPUT   OF F52      TYPE LIQUID
                  OUTPUT   OF F52      IS PRODUCT 5           TYPE LIQUID
# Connect C4 Reboiler
                  OUTPUT 2 OF C4       IS INPUT   OF C4REBL   TYPE LIQUID
                  OUTPUT 1 OF C4REBL   IS INPUT 1 OF C4       TYPE VAPOUR
  STREAM LT_DIST  OUTPUT 2 OF C4REBL   IS INPUT   OF F42      TYPE LIQUID
                  OUTPUT   OF F42      IS PRODUCT 4           TYPE LIQUID
****
UNIT C1BT29 IS A BTRAY_SS
****
UNIT C1COND IS A PCOND_SS
****
UNIT C1FT14 IS A FTRAY_SS
****
UNIT C1R IS A SECTION_SS
  SET
    TOP_TRAY = 15, BOTTOM_TRAY = 28
****
UNIT C1REBL IS A HEAT_COOL
****
UNIT C1S IS A SECTION_SS
  SET
    TOP_TRAY = 2, BOTTOM_TRAY = 13
****
UNIT C1SPL1 IS A RSPLIT
****
UNIT C1TR1 IS A TEAR
****
```

```
UNIT C2BT24 IS A BTRAY_SS
****
UNIT C2COND IS A TCOND_SS
****
UNIT C2FT14 IS A FTRAY_SS
****
UNIT C2FT17 IS A FTRAY_SS
****
UNIT C2FT21 IS A FTRAY_SS
****
UNIT C2R IS A SECTION_SS
  SET
    TOP_TRAY = 22, BOTTOM_TRAY = 23
****
UNIT C2REBL IS A HEAT_COOL
****
UNIT C2S1 IS A SECTION_SS
  SET
    TOP_TRAY = 2, BOTTOM_TRAY = 13
****
UNIT C2S3 IS A SECTION_SS
  SET
    TOP_TRAY = 19, BOTTOM_TRAY = 20
****
UNIT C2SPL1 IS A RSPLIT
****
UNIT C2SPL2 IS A RSPLIT
****
UNIT C2T15 IS A TRAY_SS
****
UNIT C2T16 IS A TRAY_SS
****
UNIT C2T18 IS A TRAY_SS
****
UNIT C2TR1 IS A TEAR
****
UNIT C2TR2 IS A TEAR
****
UNIT C2TR3 IS A TEAR
****
UNIT C4 IS A SECTION_SS
  SET
    TOP_TRAY = 1, BOTTOM_TRAY = 2
****
UNIT C4REBL IS A REBOILER_SS
```

```
****
UNIT C5 IS A SECTION_SS
  SET
    TOP_TRAY = 1, BOTTOM_TRAY = 2
****
UNIT C5REBL IS A REBOILER_SS
****
UNIT E1 IS A HEAT_COOL
****
UNIT F01 IS A VFLOW
****
UNIT F11 IS A VAP_VFLOW
****
UNIT F12 IS A LIQ_VFLOW
****
UNIT F13 IS A C5_VFLOW
****
UNIT F21 IS A VFLOW
****
UNIT F22 IS A VFLOW
****
UNIT F22A IS A LIQ_VFLOW
****
UNIT F31 IS A LIQ_VFLOW
****
UNIT F32 IS A LIQ_VFLOW
****
UNIT F41 IS A LIQ_VFLOW
****
UNIT F42 IS A TBP
****
UNIT F51 IS A LIQ_VFLOW
****
UNIT F52 IS A TBP
****
UNIT F61 IS A VFLOW
****
UNIT F62 IS A LIQ_VFLOW
****
UNIT M2R31 IS A M2R
****
UNIT MFEED IS A MOL_FEED
****
UNIT RVP31 IS A RVP
****
```

E.4 Additional rtodr6.speedup Code

The following SPEEDUP input code is specific to the `rtodr6.speedup` data reconciliation simulation. See section E.3 for the rest of the SPEEDUP input code used in `rtodr6.speedup`.

```
OPTIONS
#
# rtodr6.speedup (10/04/97)
#
  ROUTINES NEWTON, SUPERDAE, SRQP, NL2SOL, MA28
  EXECUTION
    PRINTLEVEL = 2
    TARGET = TERMINAL
    DEBUG = ON
    OPT_TOL = 1.0E-1
    SRQPGLBCON = 3
    SRQPVARINI = 1
    RESTOL = 1.0E-5
    CONVTEST = RES
    ABS_TOL = 1.0E-5
    REL_TOL = 1.0E-5
    MAXVARSTEP = 10.0
    MINNONZERO = 1.0E-8
    NONREDITER = 5
    ITERATIONS = 500
    DOGLEG = ON
    RANGEFRAC = 0.1
****
GLOBAL
# DATA RECONCILIATION OF THE PLANT
# Variables to be used in the data reconciliation
  VARIABLES
    *SF01, *SF11, *SF12, *SF13, *SF21, *SF22, *SF31, *SF32, *SF41,
    *SF42, *SF51, *SF52, *SF61, *SF62, *ST01, *ST12, *ST13, *ST21,
    *ST22, *ST23, *ST31, *ST32, *ST41, *ST42, *ST51, *ST52, *ST61,
    *ST62, *VF01, *VF11, *VF12, *VF13, *VF21, *VF22,
    *VF31, *VF32, *VF41, *VF42, *VF51, *VF52, *VF61, *VF62, *VT01,
    *VT12, *VT13, *VT21, *VT22, *VT23, *VT31, *VT32, *VT41, *VT42,
    *VT51, *VT52, *VT61, *VT62, *WF01, *WF11, *WF12,
    *WF13, *WF21, *WF22, *WF31, *WF32, *WF41, *WF42, *WF51, *WF52,
    *WF61, *WF62, *WT01, *WT12, *WT13, *WT21, *WT22, *WT23, *WT31,
    *WT32, *WT41, *WT42, *WT51, *WT52, *WT61, *WT62,
    *CNVF2B, EF01, EF11, EF12, EF13, EF21, EF22, EF31, EF32, EF41,
```

```
    EF42, EF51, EF52, EF61, EF62, ET01, ET12, ET13, ET21, ET22,
    ET23, ET31, ET32, ET41, ET42, ET51, ET52, ET61, ET62
# Data Reconciliation Objective Function
  MINIMIZE
    EF01 + EF11 + EF12 + EF13 + EF21 + EF22 + EF31 + EF32 + EF41 +
    EF42 + EF51 + EF52 + EF61 + EF62 + ET01 + ET12 + ET13 + ET21 +
    ET22 + ET23 + ET31 + ET32 + ET41 + ET42 + ET51 + ET52 + ET61 +
    ET62
  CONSTRAINT
    EF01 = WF01 * (((F01.STDVOLFL * CNVF2B) - SF01) / VF01)'2 ;
    EF11 = WF11 * ((F11.STDVOLFV - SF11) / VF11)'2 ;
    EF12 = WF12 * (((F12.STDVOLFL * CNVF2B) - SF12) / VF12)'2 ;
    EF13 = WF13 * (((F13.STDVOLFL * CNVF2B) - SF13) / VF13)'2 ;
    EF21 = WF21 * (((F21.STDVOLFL * CNVF2B) - SF21) / VF21)'2 ;
    EF22 = WF22 * (((F22.STDVOLFL * CNVF2B) - SF22) / VF22)'2 ;
    EF31 = WF31 * (((F31.STDVOLFL * CNVF2B) - SF31) / VF31)'2 ;
    EF32 = WF32 * (((F32.STDVOLFL * CNVF2B) - SF32) / VF32)'2 ;
    EF41 = WF41 * (((F41.STDVOLFL * CNVF2B) - SF41) / VF41)'2 ;
    EF42 = WF42 * (((F42.STDVOLFL * CNVF2B) - SF42) / VF42)'2 ;
    EF51 = WF51 * (((F51.STDVOLFL * CNVF2B) - SF51) / VF51)'2 ;
    EF52 = WF52 * (((F52.STDVOLFL * CNVF2B) - SF52) / VF52)'2 ;
    EF61 = WF61 * (((F61.STDVOLFL * CNVF2B) - SF61) / VF61)'2 ;
    EF62 = WF62 * (((F62.STDVOLFL * CNVF2B) - SF62) / VF62)'2 ;
    ET01 = WT01 * ((MFEED.T_OUT - ST01) / VT01)'2 ;
    ET12 = WT12 * (("C1S.TRAY_SS(2)".T - ST12) / VT12)'2 ;
    ET13 = WT13 * ((C1COND.T - ST13) / VT13)'2 ;
    ET21 = WT21 * ((C1BT29.TL_OUT1 - ST21) / VT21)'2 ;
    ET22 = WT22 * ((C1REBL.T_OUT - ST22) / VT22)'2 ;
    ET23 = WT23 * ((E1.T_OUT - ST23) / VT23)'2 ;
    ET31 = WT31 * (("C2S1.TRAY_SS(2)".T - ST31) / VT31)'2 ;
    ET32 = WT32 * ((C2COND.T - ST32) / VT32)'2 ;
    ET41 = WT41 * ((C2SPL1.TL_OUT1 - ST41) / VT41)'2 ;
    ET42 = WT42 * ((C4REBL.TL_OUT - ST42) / VT42)'2 ;
    ET51 = WT51 * ((C2SPL2.TL_OUT1 - ST51) / VT51)'2 ;
    ET52 = WT52 * ((C5REBL.TL_OUT - ST52) / VT52)'2 ;
    ET61 = WT61 * ((C2BT24.TL_OUT1 - ST61) / VT61)'2 ;
    ET62 = WT62 * ((C2REBL.T_OUT - ST62) /VT62)'2 ;
****
OPERATION
  SET
# Set Feed Stream Flow, Comp, Temp and Press.
    WITHIN MFEED
      Z_OUT = <0.000703680, 0.0246830, 0.0473640, 0.0387240,
               0.0535310, 0.0312790, 0.0351930, 0.0161490,
               0.0472560, 0.0435620, 0.0496490, 0.0530710,
```

```
                0.0523260, 0.0489630, 0.0444180, 0.041240,
                0.0607690, 0.0537820, 0.0598310, 0.0626870,
                0.056050, 0.0529210, >
      P_OUT = 200.0
# Set Partial Condenser conditions in C1
    WITHIN C1COND
      P = 163.0
      Q = 8.87771527360 : 1 : 50
    WITHIN C1SPL1
      L_OUT1 = 114.286875754 : 10 : 1000
# Set Stripping Section Tray conditions in C1
    WITHIN "C1S.TRAY_SS(2)"
      P = 167.0
      Q = 0
?REPEAT
    WITHIN "C1S.TRAY_SS(?(i))"
      P = "C1S.TRAY_SS(2)".P + (?(i)-2)*(6/27)
      Q = 0
?WITH i = <3:13>
# Set Feed Tray conditions in C1
    WITHIN C1FT14
      F_IN = 1487.17710 : 100 : 5000
      T_IN = 412.9547 : 100 : 700
      P = "C1S.TRAY_SS(2)".P + 12*(6/27)
      Q = 0
#  Rectifying Section Tray conditions in C1
?REPEAT
    WITHIN "C1R.TRAY_SS(?(i))"
      P = "C1S.TRAY_SS(2)".P + (?(i)-2)*(6/27)
      Q = 0
?WITH i = <15:28>
# Set Bottom Tray Conditions in C1
    WITHIN C1BT29
      P = "C1S.TRAY_SS(2)".P + 6.0
      Q = 0
      L_OUT1 = 1199.59366066 : 100 : 3000
      L_OUT2 = 2005.69940 : 200 : 5000
# Set Reboiler Conditions in C1
    WITHIN C1REBL
      P_OUT = "C1S.TRAY_SS(2)".P + 6.0
# Set Heat Exchanger Conditions in E1
    WITHIN E1
      P_OUT = 34.00
# Set Partial Condenser conditions in C2
    WITHIN C2COND
```

```
      P = 18.95810
      Q = 38.1525304313 : 1 : 80
      T_DROP = 28.0331682663 : 5 : 200
# Set Column Tray Conditions in C2
    WITHIN "C2S1.TRAY_SS(2)"
      P = 22.95810
      Q = 0
?REPEAT
    WITHIN "C2S1.TRAY_SS(?(i))"
      P = "C2S1.TRAY_SS(2)".P + (?(i)-2)*(12.7/22)
      Q = 0
?WITH i = <3:13>
    WITHIN C2FT14
      P = "C2S1.TRAY_SS(2)".P + 12*(12.7/22)
      Q = 0
    WITHIN C2T15
      P = "C2S1.TRAY_SS(2)".P + 13*(12.7/22)
      Q = 0
    WITHIN C2T16
      P = "C2S1.TRAY_SS(2)".P + 14*(12.7/22)
      Q = 0
    WITHIN C2FT17
      P = "C2S1.TRAY_SS(2)".P + 15*(12.7/22)
      Q = 0
    WITHIN C2T18
      P = "C2S1.TRAY_SS(2)".P + 16*(12.7/22)
      Q = 0
?REPEAT
    WITHIN "C2S3.TRAY_SS(?(i))"
      P = "C2S1.TRAY_SS(2)".P + (?(i)-2)*(12.7/22)
      Q = 0
?WITH i = <19:20>
    WITHIN C2FT21
      T_IN = 479.6565 : 449.6565 : 509.6565
      P = "C2S1.TRAY_SS(2)".P + 19*(12.7/22)
      Q = 0
?REPEAT
    WITHIN "C2R.TRAY_SS(?(i))"
      P = "C2S1.TRAY_SS(2)".P + (?(i)-2)*(12.7/22)
      Q = 0
?WITH i = <22:23>
# Set Bottom Tray Conditions in C2
    WITHIN C2BT24
      P = "C2S1.TRAY_SS(2)".P + 12.7
      Q = 0
```

```
      L_OUT1 = 526.158606822 : 20 : 2000
      L_OUT2 = 1327.217 : 20 : 4000
# Set Reboiler Conditions in C2
    WITHIN C2REBL
      P_OUT = "C2S1.TRAY_SS(2)".P + 12.7
# Set Sidedraw Splitters in C2
    WITHIN C2SPL1
      L_OUT1 = 477.609398224 : 20 : 3000
    WITHIN C2SPL2
      L_OUT1 = 248.210375677 : 20 : 3000
# Set C4 Tray Conditions
    WITHIN "C4.TRAY_SS(1)"
      P = "C2S1.TRAY_SS(2)".P + 6.92727
      Q = 0
    WITHIN "C4.TRAY_SS(2)"
      P = "C2S1.TRAY_SS(2)".P + 6.92727
      Q = 0
# Set C4 Reboiler Conditions
    WITHIN C4REBL
      P = "C2S1.TRAY_SS(2)".P + 6.92727
      L_OUT = 353.870545934 : 10 : 2000
# Set C5 Tray Conditions
    WITHIN "C5.TRAY_SS(1)"
      P = "C2S1.TRAY_SS(2)".P + 8.65909
      Q = 0
    WITHIN "C5.TRAY_SS(2)"
      P = "C2S1.TRAY_SS(2)".P + 8.65909
      Q = 0
# Set C5 Reboiler Conditions
    WITHIN C5REBL
      P = "C2S1.TRAY_SS(2)".P + 8.65909
      L_OUT = 132.989607904 : 10 : 2000
# Set Tear Stream Conditions
    WITHIN C1TR1
      F_SLACK = 0.0
      T_SLACK = 0.0
      P_SLACK = 0.0
      H_SLACK = 0.0
      Z_SLACK = <0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
                 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
                 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
                 0.0, 0.0, 0.0, 0.0, 0.0>
    WITHIN C2TR1
      F_SLACK = 0.0
      T_SLACK = 0.0
```

```
    P_SLACK = 0.0
    H_SLACK = 0.0
    Z_SLACK = <0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
               0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
               0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
               0.0, 0.0, 0.0, 0.0, 0.0>
  WITHIN C2TR2
    F_SLACK = 0.0
    T_SLACK = 0.0
    P_SLACK = 0.0
    H_SLACK = 0.0
    Z_SLACK = <0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
               0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
               0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
               0.0, 0.0, 0.0, 0.0, 0.0>
  WITHIN C2TR3
    F_SLACK = 0.0
    T_SLACK = 0.0
    P_SLACK = 0.0
    H_SLACK = 0.0
    Z_SLACK = <0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
               0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
               0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
               0.0, 0.0, 0.0, 0.0, 0.0>
  WITHIN F01
    STDTEMP = 60.0
    STDPRES = 14.696
  WITHIN F11
    STDTEMP = 60.0
    STDPRES = 14.696
  WITHIN F12
    STDTEMP = 60.0
    STDPRES = 14.696
  WITHIN F13
    STDTEMP = 60.0
    STDPRES = 14.696
  WITHIN F21
    STDTEMP = 60.0
    STDPRES = 14.696
  WITHIN F22
    STDTEMP = 60.0
    STDPRES = 14.696
  WITHIN F22A
    STDTEMP = 60.0
    STDPRES = 14.696
```

```
    WITHIN F31
      STDTEMP = 60.0
      STDPRES = 14.696
    WITHIN RVP31
      COLDTEMP = 32.0
      WARMTEMP = 100.0
      STDPRES = 14.696
      ZAIR = <0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
              0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1>
      RHOA32 = 2.78527E-3
      RHOB = 2.44686E-3
    WITHIN F32
      STDTEMP = 60.0
      STDPRES = 14.696
    WITHIN F41
      STDTEMP = 60.0
      STDPRES = 14.696
    WITHIN F42
      STDTEMP = 60.0
      STDPRES = 14.696
    WITHIN F51
      STDTEMP = 60.0
      STDPRES = 14.696
    WITHIN F52
      STDTEMP = 60.0
      STDPRES = 14.696
    WITHIN F61
      STDTEMP = 60.0
      STDPRES = 14.696
    WITHIN F62
      STDTEMP = 60.0
      STDPRES = 14.696
    WITHIN GLOBAL
#     SET MEASURED VALUES
      SF01 = 20863.014
      SF11 = 75209.44167
      SF12 = 7024.6257
      SF13 = 884.40222
      SF21 = 31955.347
      SF22 = 19080.626
      SF31 = 1580.2698
      SF32 = 12862.473
      SF41 = 5029.6623
      SF42 = 3817.694
      SF51 = 3264.5036
```

```
      SF52 = 1847.737
      SF61 = 29933.726
      SF62 = 12250.671
      ST01 = 412.9547
      ST12 = 152.33838
      ST13 = 96.372107
      ST21 = 516.0132
      ST22 = 586.08488
      ST23 = 479.69655
      ST31 = 128.35124
      ST32 = 84.722528
      ST41 = 240.31157
      ST42 = 261.0776
      ST51 = 356.19866
      ST52 = 423.89816
      ST61 = 623.44743
      ST62 = 683.89991
#     SET VARIANCES FOR MEASURED VARAIABLES
      VF01 = 1043.1507
      VF11 = 3760.4721
      VF12 = 351.23129
      VF13 = 44.22011
      VF21 = 1597.7674
      VF22 = 954.03132
      VF31 = 79.01349
      VF32 = 643.12363
      VF41 = 251.48312
      VF42 = 190.8847
      VF51 = 163.22518
      VF52 = 92.386852
      VF61 = 1496.6863
      VF62 = 612.53356
      VT01 = 20.647735
      VT12 = 7.6269189
      VT13 = 4.8186054
      VT21 = 25.80066
      VT22 = 29.304244
      VT23 = 23.984827
      VT31 = 6.4175619
      VT32 = 4.2361264
      VT41 = 12.015579
      VT42 = 13.05388
      VT51 = 17.809933
      VT52 = 21.194908
      VT61 = 31.172371
```

```
      VT62 = 34.194995
#     SET WEIGHTS FOR DATA RECONCILIATION
      WF01 = 2.0
      WF11 = 2.0
      WF12 = 2.0
      WF13 = 2.0
      WF21 = 2.0
      WF22 = 2.0
      WF31 = 2.0
      WF32 = 2.0
      WF41 = 2.0
      WF42 = 2.0
      WF51 = 2.0
      WF52 = 2.0
      WF61 = 2.0
      WF62 = 2.0
      WT01 = 2.0
      WT12 = 2.0
      WT13 = 2.0
      WT21 = 2.0
      WT22 = 2.0
      WT23 = 2.0
      WT31 = 2.0
      WT32 = 2.0
      WT41 = 2.0
      WT42 = 2.0
      WT51 = 2.0
      WT52 = 2.0
      WT61 = 2.0
      WT62 = 2.0
#     SET UNIT CONVERSIONS
      CNVF2B = 4.274857143   # Convet CUFT/HR to BBL/DAY
  FREE
    C1FT14.F_IN     : 1500
    C1FT14.T_IN     : 420
    C1SPL1.L_OUT1   : 200
    C1COND.Q        : 10
    C1BT29.L_OUT1   : 1300
    C1BT29.L_OUT2   : 2000
    C2FT21.T_IN     : 480
    C2COND.Q        : 40
    C2COND.T_DROP   : 40
    C2BT24.L_OUT1   : 600
    C2BT24.L_OUT2   : 1500
    C2SPL1.L_OUT1   : 600
```

```
    C4REBL.L_OUT    : 450
    C2SPL2.L_OUT1   : 350
    C5REBL.L_OUT    : 230
# Initial Guesses for Some Variables
#----------------------------
 PRESET         # from RESULTS
#----------------------------
# The preset subsection has been omitted from this printout as it is
# preset automatically during the real time optimization
****
EXTERNAL
  RECEIVE
# Read in Data Reconciliation Parameters (84 variables)
    GLOBAL.SF01, GLOBAL.SF11, GLOBAL.SF12, GLOBAL.SF13, GLOBAL.SF21,
    GLOBAL.SF22, GLOBAL.SF31, GLOBAL.SF32, GLOBAL.SF41, GLOBAL.SF42,
    GLOBAL.SF51, GLOBAL.SF52, GLOBAL.SF61, GLOBAL.SF62, GLOBAL.ST01,
    GLOBAL.ST12, GLOBAL.ST13, GLOBAL.ST21, GLOBAL.ST22, GLOBAL.ST23,
    GLOBAL.ST31, GLOBAL.ST32, GLOBAL.ST41, GLOBAL.ST42, GLOBAL.ST51,
    GLOBAL.ST52, GLOBAL.ST61, GLOBAL.ST62, GLOBAL.VF01, GLOBAL.VF11,
    GLOBAL.VF12, GLOBAL.VF13, GLOBAL.VF21, GLOBAL.VF22, GLOBAL.VF31,
    GLOBAL.VF32, GLOBAL.VF41, GLOBAL.VF42, GLOBAL.VF51, GLOBAL.VF52,
    GLOBAL.VF61, GLOBAL.VF62, GLOBAL.VT01, GLOBAL.VT12, GLOBAL.VT13,
    GLOBAL.VT21, GLOBAL.VT22, GLOBAL.VT23, GLOBAL.VT31, GLOBAL.VT32,
    GLOBAL.VT41, GLOBAL.VT42, GLOBAL.VT51, GLOBAL.VT52, GLOBAL.VT61,
    GLOBAL.VT62, GLOBAL.WF01, GLOBAL.WF11, GLOBAL.WF12, GLOBAL.WF13,
    GLOBAL.WF21, GLOBAL.WF22, GLOBAL.WF31, GLOBAL.WF32, GLOBAL.WF41,
    GLOBAL.WF42, GLOBAL.WF51, GLOBAL.WF52, GLOBAL.WF61, GLOBAL.WF62,
    GLOBAL.WT01, GLOBAL.WT12, GLOBAL.WT13, GLOBAL.WT21, GLOBAL.WT22,
    GLOBAL.WT23, GLOBAL.WT31, GLOBAL.WT32, GLOBAL.WT41, GLOBAL.WT42,
    GLOBAL.WT51, GLOBAL.WT52, GLOBAL.WT61, GLOBAL.WT62,
# Read in Simulation Set Points (43 variables)
    MFEED.Z_OUT(1), MFEED.Z_OUT(2), MFEED.Z_OUT(3), MFEED.Z_OUT(4),
    MFEED.Z_OUT(5), MFEED.Z_OUT(6), MFEED.Z_OUT(7), MFEED.Z_OUT(8),
    MFEED.Z_OUT(9), MFEED.Z_OUT(10), MFEED.Z_OUT(11), MFEED.Z_OUT(12),
    MFEED.Z_OUT(13), MFEED.Z_OUT(14), MFEED.Z_OUT(15), MFEED.Z_OUT(16),
    MFEED.Z_OUT(17), MFEED.Z_OUT(18), MFEED.Z_OUT(19), MFEED.Z_OUT(20),
    MFEED.Z_OUT(21), MFEED.Z_OUT(22), MFEED.P_OUT, C1COND.P, C1COND.Q,
    C1SPL1.L_OUT1, "C1S.TRAY_SS(2)".P, C1FT14.F_IN, C1FT14.T_IN,
    C1BT29.L_OUT1, C1BT29.L_OUT2, E1.P_OUT, C2COND.P, C2COND.Q,
    C2COND.T_DROP, "C2S1.TRAY_SS(2)".P, C2FT21.T_IN, C2BT24.L_OUT1,
    C2BT24.L_OUT2, C2SPL1.L_OUT1, C2SPL2.L_OUT1, C4REBL.L_OUT,
    C5REBL.L_OUT,
# Read in Free variables Bounds (30 variables)
    UBOUND (C1COND.Q), LBOUND(C1COND.Q),
    UBOUND (C1SPL1.L_OUT1), LBOUND (C1SPL1.L_OUT1),
```

```
    UBOUND (C1FT14.F_IN), LBOUND (C1FT14.F_IN),
    UBOUND (C1FT14.T_IN), LBOUND (C1FT14.T_IN),
    UBOUND (C1BT29.L_OUT1), LBOUND (C1BT29.L_OUT1),
    UBOUND (C1BT29.L_OUT2), LBOUND (C1BT29.L_OUT2),
    UBOUND (C2COND.Q), LBOUND (C2COND.Q),
    UBOUND (C2COND.T_DROP), LBOUND (C2COND.T_DROP),
    UBOUND (C2FT21.T_IN), LBOUND (C2FT21.T_IN),
    UBOUND (C2BT24.L_OUT1), LBOUND (C2BT24.L_OUT1),
    UBOUND (C2BT24.L_OUT2), LBOUND (C2BT24.L_OUT2),
    UBOUND (C2SPL1.L_OUT1), LBOUND (C2SPL1.L_OUT1),
    UBOUND (C2SPL2.L_OUT1), LBOUND (C2SPL2.L_OUT1),
    UBOUND (C4REBL.L_OUT), LBOUND (C4REBL.L_OUT),
    UBOUND (C5REBL.L_OUT), LBOUND (C5REBL.L_OUT)
  TRANSMIT
# Write out Simulations Set Points after Data Rec (44 variables)
    MFEED.Z_OUT(1), MFEED.Z_OUT(2), MFEED.Z_OUT(3), MFEED.Z_OUT(4),
    MFEED.Z_OUT(5), MFEED.Z_OUT(6), MFEED.Z_OUT(7), MFEED.Z_OUT(8),
    MFEED.Z_OUT(9), MFEED.Z_OUT(10), MFEED.Z_OUT(11), MFEED.Z_OUT(12),
    MFEED.Z_OUT(13), MFEED.Z_OUT(14), MFEED.Z_OUT(15), MFEED.Z_OUT(16),
    MFEED.Z_OUT(17), MFEED.Z_OUT(18), MFEED.Z_OUT(19), MFEED.Z_OUT(20),
    MFEED.Z_OUT(21), MFEED.Z_OUT(22), MFEED.P_OUT, C1COND.P, C1COND.Q,
    C1SPL1.L_OUT1, "C1S.TRAY_SS(2)".P, C1FT14.F_IN, C1FT14.T_IN,
    C1BT29.L_OUT1, C1BT29.L_OUT2, E1.P_OUT, C2COND.P, C2COND.Q,
    C2COND.T_DROP, "C2S1.TRAY_SS(2)".P, C2FT21.T_IN, C2BT24.L_OUT1,
    C2BT24.L_OUT2, C2SPL1.L_OUT1, C2SPL2.L_OUT1, C4REBL.L_OUT,
    C5REBL.L_OUT, F31.TL_OUT
****
```

E.5 rtodr6.f

The following FORTRAN input code, `rtodr6.f`, contains SPEEDUP EDI (External Data Interface) subroutines used by `rtodr6.speedup`.

```
C RTODR6.SPEEDUP ROUTINES FOR EDI
C
C **** EXTINI ****
C
C THIS ROUTINE IS USED TO ALERT THE EXTERNAL SYSTEM
C IFLAG=0 FOR SUCCESFUL RUN
C
      SUBROUTINE EXTINI(MODE,IFLAG)
      INTEGER MODE, IFLAG
C
```

```
      WRITE (*,*) '>>>RTODR6: EXTINI CALLED - NOTHING DONE'
      IFLAG=0
      RETURN
      END
C
C **** EXTGET ****
C
C THIS ROUTINE GETS THE DATA
C IFLAG=0 FOR SUCCESSFUL RUN
C
      SUBROUTINE EXTGET(N,TAGNAM,X,T,TNEXT,IFLAG)
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      DIMENSION X(N)
      CHARACTER * 40 TAGNAM(N)
      LOGICAL THERE
      INTEGER FNFREE
C GET A FREE CHANNEL FOR THE FILE
      ICHAN=FNFREE()
C
      WRITE (*,*) '>>>RTODR6: EXTGET CALLED - READING IN DATA'
C     READ IN MEASURED VALUES FROM FILE IMEAS.DAT
      INQUIRE (FILE='IMEAS.DAT', EXIST=THERE)
      IF (.NOT.THERE) THEN
        GO TO 915
      ENDIF
      OPEN (UNIT=ICHAN, STATUS='OLD', ERR=910, FILE='IMEAS.DAT')
      REWIND (UNIT=ICHAN)
      DO 20 M=1,28
        READ (ICHAN,100,ERR=900,END=905) X(M)
   20 CONTINUE
      CLOSE (UNIT=ICHAN)
C     READ IN VARIANCES FROM FILE IVARNC.DAT
      INQUIRE (FILE='IVARNC.DAT', EXIST=THERE)
      IF (.NOT.THERE) THEN
        GO TO 920
      ENDIF
      OPEN (UNIT=ICHAN, STATUS='OLD', ERR=910, FILE='IVARNC.DAT')
      REWIND (UNIT=ICHAN)
      DO 25 M=29,56
        READ (ICHAN,100,ERR=900,END=905) X(M)
   25 CONTINUE
      CLOSE (UNIT=ICHAN)
C     READ IN WEIGHTS FROM FILE IWGHT.DAT
      INQUIRE (FILE='IWGHT.DAT', EXIST=THERE)
      IF (.NOT.THERE) THEN
```

```
        GO TO 925
      ENDIF
      OPEN (UNIT=ICHAN, STATUS='OLD', ERR=910, FILE='IWGHT.DAT')
      REWIND (UNIT=ICHAN)
      DO 30 M=57,84
        READ (ICHAN,100,ERR=900,END=905) X(M)
   30 CONTINUE
      CLOSE (UNIT=ICHAN)
C     READ IN SET VARAIBLES FROM FILE ISET.DAT
      INQUIRE (FILE='ISET.DAT', EXIST=THERE)
      IF (.NOT.THERE) THEN
        GO TO 930
      ENDIF
      OPEN (UNIT=ICHAN, STATUS='OLD', ERR=910, FILE='ISET.DAT')
      REWIND (UNIT=ICHAN)
      DO 35 M=85,127
        READ (ICHAN,105,ERR=900,END=905) X(M)
   35 CONTINUE
      CLOSE (UNIT=ICHAN)
C     READ IN FREE VARAIBLE BOUNDS FROM FILE IDFREE.DAT
      INQUIRE (FILE='IDFREE.DAT', EXIST=THERE)
      IF (.NOT.THERE) THEN
        GO TO 935
      ENDIF
      OPEN (UNIT=ICHAN, STATUS='OLD', ERR=910, FILE='IDFREE.DAT')
      REWIND (UNIT=ICHAN)
      READ(ICHAN,100,ERR=900,END=905) TEMP
      X(128) = X(109) + TEMP
      X(129) = X(109) - TEMP
      READ(ICHAN,100,ERR=900,END=905) TEMP
      X(130) = X(110) + TEMP
      X(131) = X(110) - TEMP
      READ(ICHAN,100,ERR=900,END=905) TEMP
      X(132) = X(112) + TEMP
      X(133) = X(112) - TEMP
      READ(ICHAN,100,ERR=900,END=905) TEMP
      X(134) = X(113) + TEMP
      X(135) = X(113) - TEMP
      READ(ICHAN,100,ERR=900,END=905) TEMP
      X(136) = X(114) + TEMP
      X(137) = X(114) - TEMP
      READ(ICHAN,100,ERR=900,END=905) TEMP
      X(138) = X(115) + TEMP
      X(139) = X(115) - TEMP
      READ(ICHAN,100,ERR=900,END=905) TEMP
```

```
      X(140) = X(118) + TEMP
      X(141) = X(118) - TEMP
      READ(ICHAN,100,ERR=900,END=905) TEMP
      X(142) = X(119) + TEMP
      X(143) = X(119) - TEMP
      READ(ICHAN,100,ERR=900,END=905) TEMP
      X(144) = X(121) + TEMP
      X(145) = X(121) - TEMP
      READ(ICHAN,100,ERR=900,END=905) TEMP
      X(146) = X(122) + TEMP
      X(147) = X(122) - TEMP
      READ(ICHAN,100,ERR=900,END=905) TEMP
      X(148) = X(123) + TEMP
      X(149) = X(123) - TEMP
      READ(ICHAN,100,ERR=900,END=905) TEMP
      X(150) = X(124) + TEMP
      X(151) = X(124) - TEMP
      READ(ICHAN,100,ERR=900,END=905) TEMP
      X(152) = X(125) + TEMP
      X(153) = X(125) - TEMP
      READ(ICHAN,100,ERR=900,END=905) TEMP
      X(154) = X(126) + TEMP
      X(155) = X(126) - TEMP
      READ(ICHAN,100,ERR=900,END=905) TEMP
      X(156) = X(127) + TEMP
      X(157) = X(127) - TEMP
      CLOSE (UNIT=ICHAN)
      WRITE (*,*) '>>>RTODR6: EXTGET - DATA READ SUCCESSFULLY'
      IFLAG=0
  100 FORMAT(15X, E15.8)
  105 FORMAT(42X, E15.8)
      RETURN
C ERROR STATEMENTS
  900 WRITE(*,*) 'RTODR6: EXTGET: Error reading ',TAGNAM(M)
      IFLAG = -1
      RETURN
  905 WRITE(*,*) 'RTODR6: EXTGET: EOF reached reading ',TAGNAM(M)
      IFLAG = -1
      RETURN
  910 WRITE(*,*) 'RTODR6: EXTGET: Error opening file'
      IFLAG = -1
      RETURN
  915 WRITE(*,*) 'RTODR6: EXTGET: IMEAS.DAT does not exist'
      IFLAG = -1
      RETURN
```

```
  920 WRITE(*,*) 'RTODR6: EXTGET: IVARNC.DAT does not exist'
      IFLAG = -1
      RETURN
  925 WRITE(*,*) 'RTODR6: EXTGET: IWGHT.DAT does not exist'
      IFLAG = -1
      RETURN
  930 WRITE(*,*) 'RTODR6: EXTGET: ISET.DAT does not exist'
      IFLAG = -1
      RETURN
  935 WRITE(*,*) 'RTODR6: EXTGET: IDFREE.DAT does not exist'
      IFLAG = -1
      RETURN
      END
C
C **** EXTPUT ****
C
C THIS ROUTINE WRITES OUT THE CALCULATED VARIABLES
C IFLAG=0 FOR SUCCESFUL RUN
C
      SUBROUTINE EXTPUT(N,TAGNAM,X,T,IFLAG)
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      DIMENSION X(N)
      CHARACTER * 40 TAGNAM(N)
      INTEGER FNFREE
      LOGICAL THERE
C GET A FREE CHANNEL FOR THE FILE
      ICHAN=FNFREE()
C
      WRITE (*,*) '>>>RTODR6: EXTPUT CALLED - WRITING OUT RESULTS'
C     WRITE OUT SET VARAIBLES TO FILE RDSET.DAT
      INQUIRE (FILE='RDSET.DAT', EXIST=THERE)
      IF (THERE) THEN
        GO TO 915
      ENDIF
      OPEN (UNIT=ICHAN, STATUS='NEW', ERR=910, FILE='RDSET.DAT')
      REWIND (UNIT=ICHAN)
      DO 10 M=1,44
        WRITE(ICHAN,100,ERR=900) TAGNAM(M), X(M)
   10 CONTINUE
      CLOSE (UNIT=ICHAN)
      WRITE (*,*) '>>>RTODR6: EXTPUT - DATA WRITTEN SUCCESSFULLY'
      IFLAG=0
  100 FORMAT(A40, 2X, E15.8)
      RETURN
  900 WRITE(*,*) 'RTODR6: EXTPUT: Error writing ', TAGNAM(M)
```

```
      IFLAG = -1
      RETURN
  910 WRITE(*,*) 'RTODR6: EXTPUT: Error opening file'
      IFLAG = -1
      RETURN
  915 WRITE(*,*) 'RTODR6: EXTPUT: RDSET.DAT already exists'
      IFLAG = -1
      RETURN
      END
C
C **** EXTTRM ****
C
C TERMINATION PROGRAMME
C IFLAG=0 FOR SUCCESFUL RUN
C
      SUBROUTINE EXTTRM(IFLAG)
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
C
      WRITE (*,*)'>>>RTODR6: EXTINI CALLED - EDI FINISHED'
      IFLAG=0
      RETURN
      END
C
C **** EXTWAI ****
C
C WAIT PROGRAMME
C IFLAG=0 CONTINUE RUNNING
C IFLAG=-1 REQUEST TO TERMINATE
C
      SUBROUTINE EXTWAI (T,IFLAG)
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
C
      WRITE (*,*) '>>>RTODR6: EXTWAI CALLED - TERMINATING RUN'
      IFLAG =-1
      RETURN
      END
C
C **** EXTMSG ****
C
C MESSAGE PROGRAMME
C IFLAG=0 FOR SUCCESSFUL RUN
C
      SUBROUTINE EXTMSG (MESSGE, MESTYP, IFLAG )
      CHARACTER * (*) MESSGE
      INTEGER MESSTYP, IFLAG
```

```
      IFLAG=0
      WRITE (*,*) '>>>RTODR6: EXTMSG CALLED - WRITING MESSAGE '
      WRITE (*,*) 'MESSAGE OF TYPE',MESTYP
      WRITE (*,*) MESSGE
      WRITE (*,*) 'RTODR6: EXTMSG - MESSAGE HAS BEEN WRITTEN'
      RETURN
      END
C
C FUNCTION TO FIND A FREE TERMINAL
C
         INTEGER FUNCTION FNFREE()
C        COPYRIGHT (C) ASPENTECH UK LTD, 1991

C        FNFREE FINDS THE FIRST FREE LOGICAL UNIT FOR ASSIGNING TO A FILE.
C        IT DOES THIS BY TRYING THEM ALL IN ASCENDING ORDER UNTIL ONE IS FREE.
C        RETURNS -1 IF NONE FREE.

         IMPLICIT NONE
         LOGICAL OPEND
         INTEGER I
         DO 100 I=1,119
          INQUIRE(UNIT=I,OPENED=OPEND)
          IF (.NOT.OPEND) THEN
           FNFREE=I
           RETURN
          ENDIF
 100     CONTINUE
         FNFREE=-1
         RETURN
         END
```

E.6 Additional rtopt6.speedup Code

The following SPEEDUP input code is specific to the `rtopt6.speedup` optimization simulation. See section E.3 for the rest of the SPEEDUP input code used in `rtopt6.speedup`.

```
OPTIONS
#
# rtopt6.speedup (10/04/97)
#
  ROUTINES NEWTON, SUPERDAE, SRQP, NL2SOL, MA28
```

```
  EXECUTION
    PRINTLEVEL = 2
    TARGET = TERMINAL
    DEBUG = ON
    OPT_TOL = 1.0E-1
    SRQPGLBCON = 2
    SRQPVARINI = 1
    RESTOL = 1.0E-5
    CONVTEST = RES
    ABS_TOL = 1.0E-5
    REL_TOL = 1.0E-5
    MAXVARSTEP = 10.0
    MINNONZERO = 1.0E-8
    NONREDITER = 5
    ITERATIONS = 500
    DOGLEG = ON
    RANGEFRAC = 0.1
****
GLOBAL
# OPTMISATION OF THE PLANT TO MAXIMISE PROFIT
# Variables to be used in the optimization
  VARIABLES
    QLTISO, *PRMRVP, *SPCRVP, *DLTISO, *DVAP, *DDIST, *DHVISO,
    *DLTDST, *DSSFED, *DSTAB, *DFO, *CNVF2M, *CNVBTM, *CNVBTU,
    *CNVP2K, *F1EFF, *F2EFF, PVALS, CSTAB, COF1, COF2, PVAP, PDIST,
    PLTISO, PHVISO, PLTDST, PSSFED, *CNVF2B, *WOBJ,
    *A11U, *A11L, *A31UR, *A31LT, *A41U1, *A41L1, *A41U9, *A41L9,
    *A51U1, *A51L1, *A51U9, *A51L9, *F12U, *F12L, *F32U, *F41U,
    *F51U, *T11U, *T11L, *T22U, *T22L, *T31U, *T31L, *T62U, *T62L,
    *XF13L, *XF13U, *XF21L, *XF21U, *XF22L, *XF22U, *XT23L, *XT23U,
    *XF61L, *XF61U, *XF62L, *XF62U, *XT32L, *XT32U, *XF42L, *XF42U,
    *XF52L, *XF52U, *XF41L, *XF41U, *XF51L, *XF51U
# Economic Objective Function
  MAXIMIZE
    (PVALS - CSTAB - COF1 - COF2) * WOBJ
  CONSTRAINT
# Equality Constraints
#   Premiums [($/M3)=($/M3)+($/M3/UNIT)*(UNIT-UNIT)]
    QLTISO = DLTISO + (PRMRVP * ((RVP31.RVP * CNVP2K) - SPCRVP)) ;
#   Product Values [($/HR)=($/M3)*(M3 HR/HR FT3)*(FT3/HR)]
    PVAP = DVAP * CNVF2M * F11.STDVOLFL ;
    PDIST = DDIST * CNVF2M * F13.STDVOLFL ;
    PLTISO = QLTISO * CNVF2M * F31.STDVOLFL ;
    PHVISO = DHVISO * CNVF2M * F42.STDVOLFL ;
    PLTDST = DLTDST * CNVF2M * F52.STDVOLFL ;
```

```
    PSSFED = DSSFED * CNVF2M * F62.STDVOLFL ;
    PVALS = PVAP + PDIST + PLTISO + PHVISO + PLTDST + PSSFED ;
#   Feed Cost [($/HR)=($/M3)*(M3 HR/HR FT3)*(FT3/HR)]
    CSTAB = DSTAB * CNVF2M * FO1.STDVOLFL ;
#   Utility Costs [($/HR)=($/LFEM3)*(LFEM3/LFEBBL)*(LFEBBL/MMBTU)
#                         *(MMBTU/HR)*(1/FEFF)]
    COF1 = DFO * CNVBTM * CNVBTU * C1REBL.Q * (1 / F1EFF) ;
    COF2 = DFO * CNVBTM * CNVBTU * C2REBL.Q * (1 / F2EFF) ;
# Inequality Constraints
    F13.C5S >= A11L ;
    F13.C5S <= A11U ;
    F42.TBP10 >= A41L1 ;
    F42.TBP10 <= A41U1 ;
    F42.TBP90 >= A41L9 ;
    F42.TBP90 <= A41U9 ;
    F52.TBP10 >= A51L1 ;
    F52.TBP10 <= A51U1 ;
    F52.TBP90 >= A51L9 ;
    F52.TBP90 <= A51U9 ;
    F12.STDVOLFL >= F12L ;
    F12.STDVOLFL <= F12U ;
    F32.STDVOLFL <= F32U ;
    F41.STDVOLFL <= F41U ;
    F51.STDVOLFL <= F51U ;
    RVP31.RVP <= A31UR ;
    C2COND.T_BUBBLE >= A31LT ;
    "C1S.TRAY_SS(3)".T >= T11L ;
    "C1S.TRAY_SS(3)".T <= T11U ;
    C1REBL.T_OUT >= T22L ;
    C1REBL.T_OUT <= T22U ;
    "C2S1.TRAY_SS(2)".T >= T31L ;
    "C2S1.TRAY_SS(2)".T <= T31U ;
    C2REBL.T_OUT >= T62L ;
    C2REBL.T_OUT <= T62U ;
# Indirect Free Variable Constraints
    F13.L_OUT >= XF13L ;
    F13.L_OUT <= XF13U ;
    C1TR1.F_IN >= XF21L ;
    C1TR1.F_IN <= XF21U ;
    E1.F_IN >= XF22L ;
    E1.F_IN <= XF22U ;
    E1.T_OUT >= XT23L ;
    E1.T_OUT <= XT23U ;
    C2TR3.F_IN >= XF61L ;
    C2TR3.F_IN <= XF61U ;
```

```
    F62.L_OUT >= XF62L ;
    F62.L_OUT <= XF62U ;
    F31.TL_OUT >= XT32L ;
    F31.TL_OUT <= XT32U ;
    F42.L_OUT >= XF42L ;
    F42.L_OUT <= XF42U ;
    F52.L_OUT >= XF52L ;
    F52.L_OUT <= XF52U ;
    "C4.TRAY_SS(1)".L_IN >= XF41L ;
    "C4.TRAY_SS(1)".L_IN <= XF41U ;
    "C5.TRAY_SS(1)".L_IN >= XF51L ;
    "C5.TRAY_SS(1)".L_IN <= XF51U ;
****
OPERATION
  SET
# Set Feed Stream Flow, Comp, Temp and Press.
    WITHIN MFEED
      Z_OUT = <0.000703680, 0.0246830, 0.0473640, 0.0387240,
               0.0535310, 0.0312790, 0.0351930, 0.0161490,
               0.0472560, 0.0435620, 0.0496490, 0.0530710,
               0.0523260, 0.0489630, 0.0444180, 0.041240,
               0.0607690, 0.0537820, 0.0598310, 0.0626870,
               0.056050, 0.0529210, >
      P_OUT = 200.0
# Set Partial Condenser conditions in C1
    WITHIN C1COND
      P = 163.0
      Q = 10.28432
    WITHIN C1SPL1
      L_OUT1 = 118.0199 : 10 : 1000
# Set Rectifying Section Tray conditions in C1
    WITHIN "C1S.TRAY_SS(2)"
      P = 167.0
      Q = 0
?REPEAT
    WITHIN "C1S.TRAY_SS(?(i))"
      P = "C1S.TRAY_SS(2)".P + (?(i)-2)*(6/27)
      Q = 0
?WITH i = <3:13>
# Set Feed Tray conditions in C1
    WITHIN C1FT14
      F_IN = 1464.1499
      T_IN = 430.4696
      P = "C1S.TRAY_SS(2)".P + 12*(6/27)
      Q = 0
```

```
#  Stripping Section Tray conditions in C1
?REPEAT
    WITHIN "C1R.TRAY_SS(?(i))"
      P = "C1S.TRAY_SS(2)".P + (?(i)-2)*(6/27)
      Q = 0
?WITH i = <15:28>
# Set Bottom Tray Conditions in C1
    WITHIN C1BT29
      P = "C1S.TRAY_SS(2)".P + 6.0
      Q = 0
      L_OUT1 = 1175.6446 : 100 : 4000
      L_OUT2 = 2004.8814 : 100 : 5000
# Set Reboiler Conditions in C1
    WITHIN C1REBL
      P_OUT = 167.0 + 6.0
# Set Heat Exchanger Conditions in E1
    WITHIN E1
      P_OUT = "C1S.TRAY_SS(2)".P + 6.0
# Set Partial Condenser conditions in C2
    WITHIN C2COND
      P = 18.95810
      Q = 25.4106
      T_DROP = 32.9897 : 5 : 200
# Set Column Tray Conditions in C2
    WITHIN "C2S1.TRAY_SS(2)"
      P = 22.95810
      Q = 0
?REPEAT
    WITHIN "C2S1.TRAY_SS(?(i))"
      P = "C2S1.TRAY_SS(2)".P + (?(i)-2)*(12.7/22)
      Q = 0
?WITH i = <3:13>
    WITHIN C2FT14
      P = "C2S1.TRAY_SS(2)".P + 12*(12.7/22)
      Q = 0
    WITHIN C2T15
      P = "C2S1.TRAY_SS(2)".P + 13*(12.7/22)
      Q = 0
    WITHIN C2T16
      P = "C2S1.TRAY_SS(2)".P + 14*(12.7/22)
      Q = 0
    WITHIN C2FT17
      P = "C2S1.TRAY_SS(2)".P + 15*(12.7/22)
      Q = 0
    WITHIN C2T18
```

```
      P = "C2S1.TRAY_SS(2)".P + 16*(12.7/22)
      Q = 0
?REPEAT
    WITHIN "C2S3.TRAY_SS(?(i))"
      P = "C2S1.TRAY_SS(2)".P + (?(i)-2)*(12.7/22)
      Q = 0
?WITH i = <19:20>
    WITHIN C2FT21
      T_IN = 449.6569 : 100 : 700
      P = "C2S1.TRAY_SS(2)".P + 19*(12.7/22)
      Q = 0
?REPEAT
    WITHIN "C2R.TRAY_SS(?(i))"
      P = "C2S1.TRAY_SS(2)".P + (?(i)-2)*(12.7/22)
      Q = 0
?WITH i = <22:23>
# Set Bottom Tray Conditions in C2
    WITHIN C2BT24
      P = "C2S1.TRAY_SS(2)".P + 12.7
      Q = 0
      L_OUT1 = 514.2549 : 50 : 4000
      L_OUT2 = 1328.1557 : 500 : 3000
# Set Reboiler Conditions in C2
    WITHIN C2REBL
      P_OUT = "C2S1.TRAY_SS(2)".P + 12.7
# Set Sidedraw Splitters in C2
    WITHIN C2SPL1
      L_OUT1 = 477.2064 : 50 : 2000
    WITHIN C2SPL2
      L_OUT1 = 241.6230 : 50 : 2000
# Set C4 Tray Conditions
    WITHIN "C4.TRAY_SS(1)"
      P = "C2S1.TRAY_SS(2)".P + 6.92727
      Q = 0
    WITHIN "C4.TRAY_SS(2)"
      P = "C2S1.TRAY_SS(2)".P + 6.92727
      Q = 0
# Set C4 Reboiler Conditions
    WITHIN C4REBL
      P = "C2S1.TRAY_SS(2)".P + 6.92727
      L_OUT = 353.7547 : 20 : 2000
# Set C5 Tray Conditions
    WITHIN "C5.TRAY_SS(1)"
      P = "C2S1.TRAY_SS(2)".P + 8.65909
      Q = 0
```

```
    WITHIN "C5.TRAY_SS(2)"
      P = "C2S1.TRAY_SS(2)".P + 8.65909
      Q = 0
# Set C5 Reboiler Conditions
    WITHIN C5REBL
      P = "C2S1.TRAY_SS(2)".P + 8.65909
      L_OUT = 131.1071 : 10 : 2000
# Set Tear Stream Conditions
    WITHIN C1TR1
      F_SLACK = 0.0
      T_SLACK = 0.0
      P_SLACK = 0.0
      H_SLACK = 0.0
      Z_SLACK = <0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
                 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
                 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
                 0.0, 0.0, 0.0, 0.0, 0.0>
    WITHIN C2TR1
      F_SLACK = 0.0
      T_SLACK = 0.0
      P_SLACK = 0.0
      H_SLACK = 0.0
      Z_SLACK = <0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
                 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
                 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
                 0.0, 0.0, 0.0, 0.0, 0.0>
    WITHIN C2TR2
      F_SLACK = 0.0
      T_SLACK = 0.0
      P_SLACK = 0.0
      H_SLACK = 0.0
      Z_SLACK = <0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
                 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
                 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
                 0.0, 0.0, 0.0, 0.0, 0.0>
    WITHIN C2TR3
      F_SLACK = 0.0
      T_SLACK = 0.0
      P_SLACK = 0.0
      H_SLACK = 0.0
      Z_SLACK = <0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
                 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
                 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
                 0.0, 0.0, 0.0, 0.0, 0.0>
    WITHIN F01
```

```
  STDTEMP = 60.0
  STDPRES = 14.696
WITHIN F11
  STDTEMP = 60.0
  STDPRES = 14.696
WITHIN F12
  STDTEMP = 60.0
  STDPRES = 14.696
WITHIN F13
  STDTEMP = 60.0
  STDPRES = 14.696
WITHIN F21
  STDTEMP = 60.0
  STDPRES = 14.696
WITHIN F22
  STDTEMP = 60.0
  STDPRES = 14.696
WITHIN F22A
  STDTEMP = 60.0
  STDPRES = 14.696
WITHIN F31
  STDTEMP = 60.0
  STDPRES = 14.696
WITHIN RVP31
  COLDTEMP = 32.0
  WARMTEMP = 100.0
  STDPRES = 14.696
  ZAIR = <0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
          0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1>
  RHOA32 = 2.78527E-3
  RHOB = 2.44686E-3
WITHIN F32
  STDTEMP = 60.0
  STDPRES = 14.696
WITHIN F41
  STDTEMP = 60.0
  STDPRES = 14.696
WITHIN F42
  STDTEMP = 60.0
  STDPRES = 14.696
WITHIN F51
  STDTEMP = 60.0
  STDPRES = 14.696
WITHIN F52
  STDTEMP = 60.0
```

```
      STDPRES = 14.696
    WITHIN F61
      STDTEMP = 60.0
      STDPRES = 14.696
    WITHIN F62
      STDTEMP = 60.0
      STDPRES = 14.696
    WITHIN GLOBAL
#     Set Premium Values
      PRMRVP = -0.18     # Preimium for lower RVP in LT-ISO ($/M3/KPA)
      SPCRVP = 97.0      # Specified RVP Value (KPA)
#     Set Product Values ($/M3)
      DVAP = 0.15
      DDIST = 100.07
      DLTISO = 182.69
      DHVISO = 193.89
      DLTDST = 180.52
      DSSFED = 158.34
#     Set Feed Cost & Fuel Oil Cost ($/M3)
      DSTAB = 158.34
      DFO = 82.14
#     Set Furnace Efficiencies
      F1EFF = 0.70
      F2EFF = 0.70
#     Set Unit Conversions
      CNVF2M = 2.831685E-2  # Convet CUFT/HR to M3/HR
      CNVBTM = 0.1589873    # Convert BBL to M3
      CNVBTU = 0.158730158  # Convert MMBTU to LFEBBL (= 1/6.3)
      CNVP2K = 6.8948       # Convert PSI to KPA
      CNVF2B = 4.274857143  # Convet CUFT/HR to BBL/DAY
#     Set Objective Function Weight
      WOBJ = 2.0
#     Set Constraint Values
      A11U = 0.150
      A11L = 0.005
      A31UR = 14.0
      A31LT = 110.0
      A41U1 = 266.0
      A41L1 = 176.0
      A41U9 = 464.0
      A41L9 = 284.0
      A51U1 = 554.0
      A51L1 = 302.0
      A51U9 = 554.0
      A51L9 = 302.0
```

```
      F12U =  9000.0
      F12L =  1000.0
      F32U =  25000.0
      F41U =  10604.0
      F51U =  6000.0
      T11U =  220.0
      T11L =  125.0
      T22U =  675.0
      T22L =  530.0
      T31U =  250.0
      T31L =  120.0
      T62U =  700.0
      T62L =  550.0
#     Set Indirect Free Variable Constraint Values
      XF13L = 112.0199
      XF13U = 124.0199
      XF21L = 1904.8814
      XF21U = 2104.8814
      XF22L = 1115.6446
      XF22U = 1235.6446
      XT23L = 444.6569
      XT23U = 454.6569
      XF61L = 1263.1557
      XF61U = 1393.1557
      XF62L = 489.2549
      XF62U = 539.2549
      XT32L = 77.65493
      XT32U = 87.65493
      XF42L = 333.7547
      XF42U = 373.7547
      XF52L = 124.1071
      XF52U = 138.1071
      XF41L = 452.2064
      XF41U = 502.2064
      XF51L = 229.6230
      XF51U = 253.6230
  FREE
    C1SPL1.L_OUT1   : 200
    C1BT29.L_OUT1   : 1300
    C1BT29.L_OUT2   : 2000
    C2FT21.T_IN     : 480
    C2COND.T_DROP   : 40
    C2BT24.L_OUT1   : 600
    C2BT24.L_OUT2   : 1500
    C2SPL1.L_OUT1   : 600
```

```
    C4REBL.L_OUT    : 450
    C2SPL2.L_OUT1   : 350
    C5REBL.L_OUT    : 230
# Initial Guesses for Some Variables
#----------------------------
 PRESET         # from RESULTS
#----------------------------
# The preset subsection has been omitted from this printout as it is
# preset automatically during the real time optimization
****
EXTERNAL
  RECEIVE
# Read in Optimization Parameters (37 variables)
    GLOBAL.PRMRVP, GLOBAL.SPCRVP, GLOBAL.DVAP, GLOBAL.DDIST,
    GLOBAL.DLTISO, GLOBAL.DHVISU, GLOBAL.DLTDST, GLOBAL.DSSFED,
    GLOBAL.DSTAB, GLOBAL.DFO, GLOBAL.F1EFF, GLOBAL.F2EFF,
    GLOBAL.A11U, GLOBAL.A11L, GLOBAL.A31UR, GLOBAL.A31LT,
    GLOBAL.A41U1, GLOBAL.A41L1, GLOBAL.A41U9, GLOBAL.A41L9,
    GLOBAL.A51U1, GLOBAL.A51L1, GLOBAL.A51U9, GLOBAL.A51L9,
    GLOBAL.F12U, GLOBAL.F12L, GLOBAL.F32U, GLOBAL.F41U,
    GLOBAL.F51U, GLOBAL.T11U, GLOBAL.T11L, GLOBAL.T22U,
    GLOBAL.T22L, GLOBAL.T31U, GLOBAL.T31L, GLOBAL.T62U,
    GLOBAL.T62L,
# Read in Simulation Set Points (43 variables)
    MFEED.Z_OUT(1), MFEED.Z_OUT(2), MFEED.Z_OUT(3), MFEED.Z_OUT(4),
    MFEED.Z_OUT(5), MFEED.Z_OUT(6), MFEED.Z_OUT(7), MFEED.Z_OUT(8),
    MFEED.Z_OUT(9), MFEED.Z_OUT(10), MFEED.Z_OUT(11), MFEED.Z_OUT(12),
    MFEED.Z_OUT(13), MFEED.Z_OUT(14), MFEED.Z_OUT(15), MFEED.Z_OUT(16),
    MFEED.Z_OUT(17), MFEED.Z_OUT(18), MFEED.Z_OUT(19), MFEED.Z_OUT(20),
    MFEED.Z_OUT(21), MFEED.Z_OUT(22), MFEED.P_OUT, C1COND.P, C1COND.Q,
    C1SPL1.L_OUT1, "C1S.TRAY_SS(2)".P, C1FT14.F_IN, C1FT14.T_IN,
    C1BT29.L_OUT1, C1BT29.L_OUT2, E1.P_OUT, C2COND.P, C2COND.Q,
    C2COND.T_DROP, "C2S1.TRAY_SS(2)".P, C2FT21.T_IN, C2BT24.L_OUT1,
    C2BT24.L_OUT2, C2SPL1.L_OUT1, C2SPL2.L_OUT1, C4REBL.L_OUT,
    C5REBL.L_OUT,
# Read in Indirect Free variable constraint values (22 variables)
    GLOBAL.XF13U, GLOBAL.XF13L, GLOBAL.XF22U, GLOBAL.XF22L, GLOBAL.XF21U,
    GLOBAL.XF21L, GLOBAL.XT32U, GLOBAL.XT32L, GLOBAL.XT23U, GLOBAL.XT23L,
    GLOBAL.XF62U, GLOBAL.XF62L, GLOBAL.XF61U, GLOBAL.XF61L, GLOBAL.XF41U,
    GLOBAL.XF41L, GLOBAL.XF51U, GLOBAL.XF51L, GLOBAL.XF42U, GLOBAL.XF42L,
    GLOBAL.XF52U, GLOBAL.XF52L,
# Read in Free variables Bounds (22 variables)
    UBOUND (C1SPL1.L_OUT1), LBOUND (C1SPL1.L_OUT1),
    UBOUND (C1BT29.L_OUT1), LBOUND (C1BT29.L_OUT1),
    UBOUND (C1BT29.L_OUT2), LBOUND (C1BT29.L_OUT2),
```

```
    UBOUND (C2COND.T_DROP), LBOUND (C2COND.T_DROP),
    UBOUND (C2FT21.T_IN), LBOUND (C2FT21.T_IN),
    UBOUND (C2BT24.L_OUT1), LBOUND (C2BT24.L_OUT1),
    UBOUND (C2BT24.L_OUT2), LBOUND (C2BT24.L_OUT2),
    UBOUND (C2SPL1.L_OUT1), LBOUND (C2SPL1.L_OUT1),
    UBOUND (C2SPL2.L_OUT1), LBOUND (C2SPL2.L_OUT1),
    UBOUND (C4REBL.L_OUT), LBOUND (C4REBL.L_OUT),
    UBOUND (C5REBL.L_OUT), LBOUND (C5REBL.L_OUT),
# Read in Indirect Free variable bounds (22 variables)
    UBOUND (F13.L_OUT), LBOUND (F13.L_OUT),
    UBOUND (E1.F_IN), LBOUND (E1.F_IN),
    UBOUND (C1TR1.F_IN), LBOUND (C1TR1.F_IN),
    UBOUND (C2COND.T), LBOUND (C2COND.T),
    UBOUND (E1.T_OUT), LBOUND (E1.T_OUT),
    UBOUND (F62.L_OUT), LBOUND (F62.L_OUT),
    UBOUND (C2TR3.F_IN), LBOUND (C2TR3.F_IN),
    UBOUND ("C4.TRAY_SS(1)".L_IN), LBOUND ("C4.TRAY_SS(1)".L_IN),
    UBOUND ("C5.TRAY_SS(1)".L_IN), LBOUND ("C5.TRAY_SS(1)".L_IN),
    UBOUND (F42.L_OUT), LBOUND (F42.L_OUT),
    UBOUND (F52.L_OUT), LBOUND (F52.L_OUT),
# Read in Objective Function Weight (1 variable)
    GLOBAL.WOBJ
  TRANSMIT
# Write out Simulations Set Points after Data Rec (44 variables)
    MFEED.Z_OUT(1), MFEED.Z_OUT(2), MFEED.Z_OUT(3), MFEED.Z_OUT(4),
    MFEED.Z_OUT(5), MFEED.Z_OUT(6), MFEED.Z_OUT(7), MFEED.Z_OUT(8),
    MFEED.Z_OUT(9), MFEED.Z_OUT(10), MFEED.Z_OUT(11), MFEED.Z_OUT(12),
    MFEED.Z_OUT(13), MFEED.Z_OUT(14), MFEED.Z_OUT(15), MFEED.Z_OUT(16),
    MFEED.Z_OUT(17), MFEED.Z_OUT(18), MFEED.Z_OUT(19), MFEED.Z_OUT(20),
    MFEED.Z_OUT(21), MFEED.Z_OUT(22), MFEED.P_OUT, C1COND.P, C1COND.Q,
    C1SPL1.L_OUT1, "C1S.TRAY_SS(2)".P, C1FT14.F_IN, C1FT14.T_IN,
    C1BT29.L_OUT1, C1BT29.L_OUT2, E1.P_OUT, C2COND.P, C2COND.Q,
    C2COND.T_DROP, "C2S1.TRAY_SS(2)".P, C2FT21.T_IN, C2BT24.L_OUT1,
    C2BT24.L_OUT2, C2SPL1.L_OUT1, C2SPL2.L_OUT1, C4REBL.L_OUT,
    C5REBL.L_OUT, F31.TL_OUT,
# Write out Aspen Simulation Setting after Optimzation (208 variables)
    MFEED.Z_OUT(23), C2REBL.T_OUT, C2REBL.P_OUT, C2REBL.Z_OUT(1),
    C2REBL.Z_OUT(2), C2REBL.Z_OUT(3), C2REBL.Z_OUT(4), C2REBL.Z_OUT(5),
    C2REBL.Z_OUT(6), C2REBL.Z_OUT(7), C2REBL.Z_OUT(8), C2REBL.Z_OUT(9),
    C2REBL.Z_OUT(10), C2REBL.Z_OUT(11), C2REBL.Z_OUT(12), C2REBL.Z_OUT(13),
    C2REBL.Z_OUT(14), C2REBL.Z_OUT(15), C2REBL.Z_OUT(16), C2REBL.Z_OUT(17),
    C2REBL.Z_OUT(18), C2REBL.Z_OUT(19), C2REBL.Z_OUT(20), C2REBL.Z_OUT(21),
    C2REBL.Z_OUT(22), C2REBL.Z_OUT(23), F11.V_OUT, F12.L_OUT,
    C1REBL.T_OUT, C1BT29.P, C2COND.L_OUT2, C2BT24.P,
    C4REBL.Q, C5REBL.Q, C1COND.T, "C1S.TRAY_SS(2)".T,
```

```
"C1S.TRAY_SS(3)".T, "C1S.TRAY_SS(4)".T, "C1S.TRAY_SS(5)".T,
"C1S.TRAY_SS(6)".T, "C1S.TRAY_SS(7)".T, "C1S.TRAY_SS(8)".T,
"C1S.TRAY_SS(9)".T, "C1S.TRAY_SS(10)".T, "C1S.TRAY_SS(11)".T,
"C1S.TRAY_SS(12)".T, "C1S.TRAY_SS(13)".T, C1FT14.T,
"C1R.TRAY_SS(15)".T, "C1R.TRAY_SS(16)".T, "C1R.TRAY_SS(17)".T,
"C1R.TRAY_SS(18)".T, "C1R.TRAY_SS(19)".T, "C1R.TRAY_SS(20)".T,
"C1R.TRAY_SS(21)".T, "C1R.TRAY_SS(22)".T, "C1R.TRAY_SS(23)".T,
"C1R.TRAY_SS(24)".T, "C1R.TRAY_SS(25)".T, "C1R.TRAY_SS(26)".T,
"C1R.TRAY_SS(27)".T, "C1R.TRAY_SS(28)".T, C1BT29.T,
C2COND.T, "C2S1.TRAY_SS(2)".T, "C2S1.TRAY_SS(3)".T,
"C2S1.TRAY_SS(4)".T, "C2S1.TRAY_SS(5)".T, "C2S1.TRAY_SS(6)".T,
"C2S1.TRAY_SS(7)".T, "C2S1.TRAY_SS(8)".T, "C2S1.TRAY_SS(9)".T,
"C2S1.TRAY_SS(10)".T, "C2S1.TRAY_SS(11)".T, "C2S1.TRAY_SS(12)".T,
"C2S1.TRAY_SS(13)".T, C2FT14.T, C2T15.T,
C2T16.T, C2FT17.T, C2T18.T,
"C2S3.TRAY_SS(19)".T, "C2S3.TRAY_SS(20)".T, C2FT21.T,
"C2R.TRAY_SS(22)".T, "C2R.TRAY_SS(23)".T, C2BT24.T,
"C4.TRAY_SS(1)".T, "C4.TRAY_SS(1)".T, C4REBL.T,
"C5.TRAY_SS(1)".T, "C5.TRAY_SS(1)".T, C5REBL.T,
C1COND.L_OUT, "C1S.TRAY_SS(2)".L_OUT,
"C1S.TRAY_SS(3)".L_OUT, "C1S.TRAY_SS(4)".L_OUT, "C1S.TRAY_SS(5)".L_OUT,
"C1S.TRAY_SS(6)".L_OUT, "C1S.TRAY_SS(7)".L_OUT, "C1S.TRAY_SS(8)".L_OUT,
"C1S.TRAY_SS(9)".L_OUT, "C1S.TRAY_SS(10)".L_OUT, "C1S.TRAY_SS(11)".L_OUT,
"C1S.TRAY_SS(12)".L_OUT, "C1S.TRAY_SS(13)".L_OUT, C1FT14.L_OUT,
"C1R.TRAY_SS(15)".L_OUT, "C1R.TRAY_SS(16)".L_OUT, "C1R.TRAY_SS(17)".L_OUT,
"C1R.TRAY_SS(18)".L_OUT, "C1R.TRAY_SS(19)".L_OUT, "C1R.TRAY_SS(20)".L_OUT,
"C1R.TRAY_SS(21)".L_OUT, "C1R.TRAY_SS(22)".L_OUT, "C1R.TRAY_SS(23)".L_OUT,
"C1R.TRAY_SS(24)".L_OUT, "C1R.TRAY_SS(25)".L_OUT, "C1R.TRAY_SS(26)".L_OUT,
"C1R.TRAY_SS(27)".L_OUT, "C1R.TRAY_SS(28)".L_OUT,
C1COND.V_OUT, "C1S.TRAY_SS(2)".V_OUT,
"C1S.TRAY_SS(3)".V_OUT, "C1S.TRAY_SS(4)".V_OUT, "C1S.TRAY_SS(5)".V_OUT,
"C1S.TRAY_SS(6)".V_OUT, "C1S.TRAY_SS(7)".V_OUT, "C1S.TRAY_SS(8)".V_OUT,
"C1S.TRAY_SS(9)".V_OUT, "C1S.TRAY_SS(10)".V_OUT, "C1S.TRAY_SS(11)".V_OUT,
"C1S.TRAY_SS(12)".V_OUT, "C1S.TRAY_SS(13)".V_OUT, C1FT14.V_OUT,
"C1R.TRAY_SS(15)".V_OUT, "C1R.TRAY_SS(16)".V_OUT, "C1R.TRAY_SS(17)".V_OUT,
"C1R.TRAY_SS(18)".V_OUT, "C1R.TRAY_SS(19)".V_OUT, "C1R.TRAY_SS(20)".V_OUT,
"C1R.TRAY_SS(21)".V_OUT, "C1R.TRAY_SS(22)".V_OUT, "C1R.TRAY_SS(23)".V_OUT,
"C1R.TRAY_SS(24)".V_OUT, "C1R.TRAY_SS(25)".V_OUT, "C1R.TRAY_SS(26)".V_OUT,
"C1R.TRAY_SS(27)".V_OUT, "C1R.TRAY_SS(28)".V_OUT, C1BT29.V_OUT,
C2COND.V_IN, "C2S1.TRAY_SS(2)".L_OUT, "C2S1.TRAY_SS(3)".L_OUT,
"C2S1.TRAY_SS(4)".L_OUT, "C2S1.TRAY_SS(5)".L_OUT, "C2S1.TRAY_SS(6)".L_OUT,
"C2S1.TRAY_SS(7)".L_OUT, "C2S1.TRAY_SS(8)".L_OUT, "C2S1.TRAY_SS(9)".L_OUT,
"C2S1.TRAY_SS(10)".L_OUT, "C2S1.TRAY_SS(11)".L_OUT,
"C2S1.TRAY_SS(12)".L_OUT, "C2S1.TRAY_SS(13)".L_OUT, C2FT14.L_OUT,
C2T15.L_OUT, C2T16.L_OUT, C2FT17.L_OUT, C2T18.L_OUT,
```

```
    "C2S3.TRAY_SS(19)".L_OUT, "C2S3.TRAY_SS(20)".L_OUT, C2FT21.L_OUT,
    "C2R.TRAY_SS(22)".L_OUT, "C2R.TRAY_SS(23)".L_OUT,
    "C2S1.TRAY_SS(2)".V_OUT, "C2S1.TRAY_SS(3)".V_OUT,
    "C2S1.TRAY_SS(4)".V_OUT, "C2S1.TRAY_SS(5)".V_OUT, "C2S1.TRAY_SS(6)".V_OUT,
    "C2S1.TRAY_SS(7)".V_OUT, "C2S1.TRAY_SS(8)".V_OUT, "C2S1.TRAY_SS(9)".V_OUT,
    "C2S1.TRAY_SS(10)".V_OUT, "C2S1.TRAY_SS(11)".V_OUT,
    "C2S1.TRAY_SS(12)".V_OUT, "C2S1.TRAY_SS(13)".V_OUT,
    C2FT14.V_OUT, C2T15.V_OUT, C2T16.V_OUT, C2FT17.V_OUT, C2T18.V_OUT,
    "C2S3.TRAY_SS(19)".V_OUT, "C2S3.TRAY_SS(20)".V_OUT, C2FT21.V_OUT,
    "C2R.TRAY_SS(22)".V_OUT, "C2R.TRAY_SS(23)".V_OUT, C2BT24.V_OUT,
    "C4.TRAY_SS(1)".L_OUT, "C4.TRAY_SS(1)".L_OUT, C4REBL.L_OUT,
    "C4.TRAY_SS(1)".V_OUT, "C4.TRAY_SS(1)".V_OUT, C4REBL.V_OUT,
    "C5.TRAY_SS(1)".L_OUT, "C5.TRAY_SS(1)".L_OUT, C5REBL.L_OUT,
    "C5.TRAY_SS(1)".V_OUT, "C5.TRAY_SS(1)".V_OUT, C5REBL.V_OUT
****
```

E.7 rtopt6.f

The following FORTRAN input code, `rtopt6.f`, contains SPEEDUP EDI (External Data Interface) subroutines used by `rtopt6.speedup`.

```
C RTOPT6.SPEEDUP ROUTINES FOR EDI
C
C **** EXTINI ****
C
C THIS ROUTINE IS USED TO ALERT THE EXTERNAL SYSTEM
C IFLAG=0 FOR SUCCESFUL RUN
C
      SUBROUTINE EXTINI(MODE,IFLAG)
      INTEGER MODE, IFLAG
C
      WRITE (*,*) '>>>RTOPT6: EXTINI CALLED - NOTHING DONE'
      IFLAG=0
      RETURN
      END
C
C **** EXTGET ****
C
C THIS ROUTINE GETS THE DATA
C IFLAG=0 FOR SUCCESSFUL RUN
C
      SUBROUTINE EXTGET(N,TAGNAM,X,T,TNEXT,IFLAG)
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
```

```
      DIMENSION X(N)
      CHARACTER * 40 TAGNAM(N)
      LOGICAL THERE
      INTEGER FNFREE
C GET A FREE CHANNEL FOR THE FILE
      ICHAN=FNFREE()
C
      WRITE (*,*) '>>>RTOPT6: EXTGET CALLED - READING IN DATA'
C     READ IN MEASURED VALUES FROM FILE IOBJFN.DAT
      INQUIRE (FILE='IOBJFN.DAT', EXIST=THERE)
      IF (.NOT.THERE) THEN
        GO TO 915
      ENDIF
      OPEN (UNIT=ICHAN, STATUS='OLD', ERR=910, FILE='IOBJFN.DAT')
      REWIND (UNIT=ICHAN)
      DO 20 M=1,12
        READ (ICHAN,100,ERR=900,END=905) X(M)
   20 CONTINUE
C     READ IN OBJECTIVE FUNCTION WEIGHT
      READ (ICHAN,100,ERR=900,END=905) X(147)
      CLOSE (UNIT=ICHAN)
C     READ IN VARIANCES FROM FILE ICNSTR.DAT
      INQUIRE (FILE='ICNSTR.DAT', EXIST=THERE)
      IF (.NOT.THERE) THEN
        GO TO 920
      ENDIF
      OPEN (UNIT=ICHAN, STATUS='OLD', ERR=910, FILE='ICNSTR.DAT')
      REWIND (UNIT=ICHAN)
      DO 25 M=13,37
        READ (ICHAN,100,ERR=900,END=905) X(M)
   25 CONTINUE
      CLOSE (UNIT=ICHAN)
C     READ IN SET VARAIBLES FROM FILE ISET.DAT
      INQUIRE (FILE='ISET.DAT', EXIST=THERE)
      IF (.NOT.THERE) THEN
        GO TO 930
      ENDIF
      OPEN (UNIT=ICHAN, STATUS='OLD', ERR=910, FILE='ISET.DAT')
      REWIND (UNIT=ICHAN)
      DO 35 M=38,80
        READ(ICHAN,105,ERR=900,END=905) X(M)
   35 CONTINUE
      READ(ICHAN,105,ERR=900,END=905) YF31T
      CLOSE (UNIT=ICHAN)
C     READ IN FREE VARAIBLE BOUNDS FROM FILE IOFREE.DAT
```

```
INQUIRE (FILE='IOFREE.DAT', EXIST=THERE)
IF (.NOT.THERE) THEN
  GO TO 935
ENDIF
OPEN (UNIT=ICHAN, STATUS='OLD', ERR=910, FILE='IOFREE.DAT')
REWIND (UNIT=ICHAN)
READ(ICHAN,100,ERR=900,END=905) TEMP
X(81) = X(63) + (X(63) * TEMP)
X(82) = X(63) - (X(63) * TEMP)
X(103) = X(63) + (X(63) * TEMP * 1.5)
X(104) = X(63) - (X(63) * TEMP * 1.5)
X(125) = X(63) + (X(63) * TEMP)
X(126) = X(63) - (X(63) * TEMP)
READ(ICHAN,100,ERR=900,END=905) TEMP
X(83) = X(67) + (X(67) * TEMP)
X(84) = X(67) - (X(67) * TEMP)
X(105) = X(67) + (X(67) * TEMP * 1.5)
X(106) = X(67) - (X(67) * TEMP * 1.5)
X(127) = X(67) + (X(67) * TEMP)
X(128) = X(67) - (X(67) * TEMP)
READ(ICHAN,100,ERR=900,END=905) TEMP
X(85) = X(68) + (X(68) * TEMP)
X(86) = X(68) - (X(68) * TEMP)
X(107) = X(68) + (X(68) * TEMP * 1.5)
X(108) = X(68) - (X(68) * TEMP * 1.5)
X(129) = X(68) + (X(68) * TEMP)
X(130) = X(68) - (X(68) * TEMP)
READ(ICHAN,100,ERR=900,END=905) TEMP
X(87) = YF31T + TEMP
X(88) = YF31T - TEMP
X(109) = X(72) + (TEMP * 1.5)
X(110) = X(72) - (TEMP * 1.5)
X(131) = YF31T + TEMP
X(132) = YF31T - TEMP
READ(ICHAN,100,ERR=900,END=905) TEMP
X(89) = X(74) + TEMP
X(90) = X(74) - TEMP
X(111) = X(74) + (TEMP * 1.5)
X(112) = X(74) - (TEMP * 1.5)
X(133) = X(74) + TEMP
X(134) = X(74) - TEMP
READ(ICHAN,100,ERR=900,END=905) TEMP
X(91) = X(75) + (X(75) * TEMP)
X(92) = X(75) - (X(75) * TEMP)
X(113) = X(75) + (X(75) * TEMP * 1.5)
```

```
      X(114) = X(75) - (X(75) * TEMP * 1.5)
      X(135) = X(75) + (X(75) * TEMP)
      X(136) = X(75) - (X(75) * TEMP)
      READ(ICHAN,100,ERR=900,END=905) TEMP
      X(93) = X(76) + (X(76) * TEMP)
      X(94) = X(76) - (X(76) * TEMP)
      X(115) = X(76) + (X(76) * TEMP * 1.5)
      X(116) = X(76) - (X(76) * TEMP * 1.5)
      X(137) = X(76) + (X(76) * TEMP)
      X(138) = X(76) - (X(76) * TEMP)
      READ(ICHAN,100,ERR=900,END=905) TEMP
      X(95) = X(77) + (X(77) * TEMP)
      X(96) = X(77) - (X(77) * TEMP)
      X(117) = X(77) + (X(77) * TEMP * 1.5)
      X(118) = X(77) - (X(77) * TEMP * 1.5)
      X(139) = X(77) + (X(77) * TEMP)
      X(140) = X(77) - (X(77) * TEMP)
      READ(ICHAN,100,ERR=900,END=905) TEMP
      X(97) = X(78) + (X(78) * TEMP)
      X(98) = X(78) - (X(78) * TEMP)
      X(119) = X(78) + (X(78) * TEMP * 1.5)
      X(120) = X(78) - (X(78) * TEMP * 1.5)
      X(141) = X(78) + (X(78) * TEMP)
      X(142) = X(78) - (X(78) * TEMP)
      READ(ICHAN,100,ERR=900,END=905) TEMP
      X(99) = X(79) + (X(79) * TEMP)
      X(100) = X(79) - (X(79) * TEMP)
      X(121) = X(79) + (X(79) * TEMP * 1.5)
      X(122) = X(79) - (X(79) * TEMP * 1.5)
      X(143) = X(79) + (X(79) * TEMP)
      X(144) = X(79) - (X(79) * TEMP)
      READ(ICHAN,100,ERR=900,END=905) TEMP
      X(101) = X(80) + (X(80) * TEMP)
      X(102) = X(80) - (X(80) * TEMP)
      X(123) = X(80) + (X(80) * TEMP * 1.5)
      X(124) = X(80) - (X(80) * TEMP * 1.5)
      X(145) = X(80) + (X(80) * TEMP)
      X(146) = X(80) - (X(80) * TEMP)
      CLOSE (UNIT=ICHAN)
      WRITE (*,*) '>>>RTOPT6: EXTGET - DATA READ SUCCESSFULLY'
      IFLAG=0
  100 FORMAT(15X, E15.8)
  105 FORMAT(42X, E15.8)
      RETURN
C ERROR STATEMENTS
```

```
  900 WRITE(*,*) 'RTOPT6: EXTGET: Error reading ',TAGNAM(M)
      IFLAG = -1
      RETURN
  905 WRITE(*,*) 'RTOPT6: EXTGET: EOF reached reading ',TAGNAM(M)
      IFLAG = -1
      RETURN
  910 WRITE(*,*) 'RTOPT6: EXTGET: Error opening file'
      IFLAG = -1
      RETURN
  915 WRITE(*,*) 'RTOPT6: EXTGET: IOBJFN.DAT does not exist'
      IFLAG = -1
      RETURN
  920 WRITE(*,*) 'RTOPT6: EXTGET: ICNSTR.DAT does not exist'
      IFLAG = -1
      RETURN
  930 WRITE(*,*) 'RTOPT6: EXTGET: ISET.DAT does not exist'
      IFLAG = -1
      RETURN
  935 WRITE(*,*) 'RTOPT6: EXTGET: IOFREE.DAT does not exist'
      IFLAG = -1
      RETURN
      END
C
C **** EXTPUT ****
C
C THIS ROUTINE WRITES OUT THE CALCULATED VARIABLES
C IFLAG=0 FOR SUCCESFUL RUN
C
      SUBROUTINE EXTPUT(N,TAGNAM,X,T,IFLAG)
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      DIMENSION X(N)
      CHARACTER * 40 TAGNAM(N)
      INTEGER FNFREE
      LOGICAL THERE
C GET A FREE CHANNEL FOR THE FILE
      ICHAN=FNFREE()
C
      WRITE (*,*) '>>>RTOPT6: EXTPUT CALLED - WRITING OUT RESULTS'
C     WRITE OUT SET VARAIBLES TO FILE ROSET.DAT
      INQUIRE (FILE='ROSET.DAT', EXIST=THERE)
      IF (THERE) THEN
        GO TO 915
      ENDIF
      OPEN (UNIT=ICHAN, STATUS='NEW', ERR=910, FILE='ROSET.DAT')
      DO 10 M=1,44
```

```
        WRITE(ICHAN,100,ERR=900) TAGNAM(M), X(M)
   10 CONTINUE
      CLOSE (UNIT=ICHAN)
C     WRITE TO FILE THE VARIABLES FOR ASPEN SIMULATION
      INQUIRE (FILE='ROSMST.DAT', EXIST=THERE)
      IF (THERE) THEN
        GO TO 920
      ENDIF
      OPEN (UNIT=ICHAN, STATUS='NEW', ERR=910, FILE='ROSMST.DAT')
C   STAB STREAM SPECIFICATIONS
      WRITE(ICHAN,105,ERR=900) 'STAB MOLEFLOW', X(28)
      WRITE(ICHAN,105,ERR=900) 'STAB TEMP', X(29)
      WRITE(ICHAN,105,ERR=900) 'STAB PRESS', X(23)
      DO 12 I=1,22
        WRITE(ICHAN,110,ERR=900) 'STAB Z', I, X(I)
   12 CONTINUE
      I = 23
      WRITE(ICHAN,110,ERR=900) 'STAB Z', I, X(45)
C   C2REBL STREAM ESTIMATIONS
      WRITE(ICHAN,105,ERR=900) 'C2REBL FLOW', X(39)
      WRITE(ICHAN,105,ERR=900) 'C2REBL TEMP', X(46)
      WRITE(ICHAN,105,ERR=900) 'C2REBL PRESS', X(47)
      DO 14 I=48,70
        J = I - 47
        WRITE(ICHAN,110,ERR=900) 'C2REBL Z', J, X(I)
   14 CONTINUE
C   E1 HEATER SPECIFICATIONS
      WRITE(ICHAN,105,ERR=900) 'E1 TEMP', X(37)
      WRITE(ICHAN,105,ERR=900) 'E1 PRESS', X(32)
C   C1 COLUMN SPECIFICATIONS
      C1RR = X(72) / (X(26) + X(71))
      C1RDV = X(71) / (X(26) + X(71))
      C1CDP = X(74) - X(27)
      WRITE(ICHAN,105,ERR=900) 'C1 RFLX RATIO', C1RR
      WRITE(ICHAN,105,ERR=900) 'C1 VAP/DIST', C1RDV
      WRITE(ICHAN,105,ERR=900) 'C1 REBL FLOW', X(31)
      WRITE(ICHAN,105,ERR=900) 'C1 REBL TEMP', X(73)
      WRITE(ICHAN,105,ERR=900) 'C1 COL P DROP', C1CDP
      WRITE(ICHAN,105,ERR=900) 'C1 TOP TRAY P', X(24)
      WRITE(ICHAN,105,ERR=900) 'C1 2ND TRAY P', X(27)
C   C2 COLUMN SPECIFICATIONS
      C2CDP = X(76) - X(36)
      WRITE(ICHAN,105,ERR=900) 'C2 DIST FLOW', X(75)
      WRITE(ICHAN,105,ERR=900) 'C2 COND SC T', X(44)
      WRITE(ICHAN,105,ERR=900) 'C2 COL P DROP', C2CDP
```

```
      WRITE(ICHAN,105,ERR=900) 'C2 TOP TRAY P', X(33)
      WRITE(ICHAN,105,ERR=900) 'C2 2ND TRAY P', X(36)
      WRITE(ICHAN,105,ERR=900) 'C2-C4 IC FLOW', X(40)
      WRITE(ICHAN,105,ERR=900) 'C2-C5 IC FLOW', X(41)
C   C4 COLUMN SPECIFICATIONS
      WRITE(ICHAN,105,ERR=900) 'C4 REBL DUTY', X(77)
C   C5 COLUMN SPECIFICATIONS
      WRITE(ICHAN,105,ERR=900) 'C5 REBL DUTY', X(78)
C   C1 TEMPERATURE PROFILE
      DO 16 I=79,107
        J = I - 78
        WRITE(ICHAN,110,ERR=900) 'C1 T, TRAY', J, X(I)
   16 CONTINUE
C   C2 TEMPERATURE PROFILE
      DO 18 I=108,131
        J = I - 107
        WRITE(ICHAN,110,ERR=900) 'C2 T, TRAY', J, X(I)
   18 CONTINUE
C   C4 TEMPERATURE PROFILE
      DO 20 I=132,134
        J = I - 131
        WRITE(ICHAN,110,ERR=900) 'C4 T, TRAY', J, X(I)
   20 CONTINUE
C   C5 TEMPERATURE PROFILE
      DO 27 I=135,137
        J = I - 134
        WRITE(ICHAN,110,ERR=900) 'C5 T, TRAY', J, X(I)
   27 CONTINUE
C   C1 LIQUID AND VAPOUR FLOW PROFILES
      C1L29 = X(30) + X(31)
      DO 30 I=138,165
        J = I - 137
        WRITE(ICHAN,110,ERR=900) 'C1 L, TRAY', J, X(I)
   30 CONTINUE
      J = 29
      WRITE(ICHAN,110,ERR=900) 'C1 L, TRAY', J, C1L29
      DO 32 I=166,194
        J = I - 165
        WRITE(ICHAN,110,ERR=900) 'C1 V, TRAY', J, X(I)
   32 CONTINUE
C   C2 LIQUID AND VAPOUR FLOW PROFILES
      C2L24 = X(38) + X(39)
      C2V1 = 0.0
      DO 34 I=195,217
        J = I - 194
```

```
        WRITE(ICHAN,110,ERR=900) 'C2 L, TRAY', J, X(I)
   34 CONTINUE
      J = 29
      WRITE(ICHAN,110,ERR=900) 'C2 L, TRAY', J, C2L24
      J = 1
      WRITE(ICHAN,110,ERR=900) 'C2 V, TRAY', J, C2V1
      DO 36 I=218,240
        J = I - 216
        WRITE(ICHAN,110,ERR=900) 'C2 V, TRAY', J, X(I)
C   C4 LIQUID AND VAPOUR FLOW PROFILES
   36 CONTINUE
      DO 38 I=241,243
        J = I - 240
        WRITE(ICHAN,110,ERR=900) 'C4 L, TRAY', J, X(I)
   38 CONTINUE
      DO 40 I=244,246
        J = I - 243
        WRITE(ICHAN,110,ERR=900) 'C4 V, TRAY', J, X(I)
C   C5 LIQUID AND VAPOUR FLOW PROFILES
   40 CONTINUE
      DO 42 I=247,249
        J = I - 246
        WRITE(ICHAN,110,ERR=900) 'C5 L, TRAY', J, X(I)
   42 CONTINUE
      DO 44 I=250,252
        J = I - 249
        WRITE(ICHAN,110,ERR=900) 'C5 V, TRAY', J, X(I)
   44 CONTINUE
      CLOSE (UNIT=ICHAN)
      WRITE (*,*) '>>>RTOPT6: EXTPUT - DATA WRITTEN SUCCESSFULLY'
      IFLAG=0
  100 FORMAT(A40, 2X, E15.8)
  105 FORMAT(A13, 2X, E15.8)
  110 FORMAT(A10, X, I2, 2X, E15.8)
      RETURN
  900 WRITE(*,*) 'RTOPT6: EXTPUT: Error writing ', TAGNAM(M)
      IFLAG = -1
      RETURN
  910 WRITE(*,*) 'RTOPT6: EXTPUT: Error opening file'
      IFLAG = -1
      RETURN
  915 WRITE(*,*) 'RTOPT6: EXTPUT: ROSET.DAT already exists'
      IFLAG = -1
      RETURN
  920 WRITE(*,*) 'RTOPT6: EXTPUT: ROSMST.DAT already exists'
```

```
      IFLAG = -1
      RETURN
      END
C
C **** EXTTRM ****
C
C TERMINATION PROGRAMME
C IFLAG=0 FOR SUCCESFUL RUN
C
      SUBROUTINE EXTTRM(IFLAG)
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
C
      WRITE (*,*)'>>>RTOPT6: EXTINI CALLED - EDI FINISHED'
      IFLAG=0
      RETURN
      END
C
C **** EXTWAI ****
C
C WAIT PROGRAMME
C IFLAG=0 CONTINUE RUNNING
C IFLAG=-1 REQUEST TO TERMINATE
C
      SUBROUTINE EXTWAI (T,IFLAG)
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
C
      WRITE (*,*) '>>>RTOPT6: EXTWAI CALLED - TERMINATING RUN'
      IFLAG =-1
      RETURN
      END
C
C **** EXTMSG ****
C
C MESSAGE PROGRAMME
C IFLAG=0 FOR SUCCESSFUL RUN
C
      SUBROUTINE EXTMSG (MESSGE, MESTYP, IFLAG )
      CHARACTER * (*) MESSGE
      INTEGER MESSTYP, IFLAG
      IFLAG=0
      WRITE (*,*) '>>>RTOPT6: EXTMSG CALLED - WRITING MESSAGE '
      WRITE (*,*) 'MESSAGE OF TYPE',MESTYP
      WRITE (*,*) MESSGE
      WRITE (*,*) 'RTOPT6: EXTMSG - MESSAGE HAS BEEN WRITTEN'
      RETURN
```

```
      END
C
C FUNCTION TO FIND A FREE TERMINAL
C
         INTEGER FUNCTION FNFREE()
C        COPYRIGHT (C) ASPENTECH UK LTD, 1991

C        FNFREE FINDS THE FIRST FREE LOGICAL UNIT FOR ASSIGNING TO A FILE.
C        IT DOES THIS BY TRYING THEM ALL IN ASCENDING ORDER UNTIL ONE IS FREE.
C        RETURNS -1 IF NONE FREE.

         IMPLICIT NONE
         LOGICAL OPEND
         INTEGER I
         DO 100 I=1,119
          INQUIRE(UNIT=I,OPENED=OPEND)
          IF (.NOT.OPEND) THEN
           FNFREE=I
           RETURN
          ENDIF
 100     CONTINUE
         FNFREE=-1
         RETURN
         END
```

E.8 model6x.speedup

The following SPEEDUP input code, `model6x.speedup`, contains the user defined models used in both `rtodr6.speedup` and `rtopt6.speedup`.

```
MODEL btray_ss
{
  Model Version 1.0A
  SPEEDUP Model Library
  Copyright, 1982-1995  Aspen Technology, Inc. All rights reserved.
     This copyright statement must not be deleted and must be
     included in any modification or adaptation of this Model.
}
CATEGORY Distillation

HELP
    Bottom tray model for steady state distillation modelling
    using a pumparound reboiler (same as FTRAY_SS without
```

```
    a V_IN stream)

    INPUT  1         :  flow from tray above      type LIQUID
    INPUT  2         :  feed to tray              type MAINSTREAM
    OUTPUT 1         :  output from tray          type VAPOUR
    OUTPUT 2         :  overflow from tray        type LIQUID
    OUTPUT 3         :  Reboiler pumparound draw  type LIQUID

    Preferred Sets   :  P, Q

    Parameters       :  nocomp
$ENDHELP

  SET
    nocomp                          # number of components

  TYPE
#   Input 1:
      L_in       AS                  flow_mol_liq
      x_in       AS ARRAY(nocomp) OF molefraction
      Tl_in      AS                  temperature
      Pl_in      AS                  pressure
      hl_in      AS                  enth_mol_liq
#   Input 2:
      F_in       AS                  flow_mol
      z_in       AS ARRAY(nocomp) OF molefraction
      T_in       AS                  temperature
      P_in       AS                  pressure
      h_in       AS                  enth_mol
#   Output 1:
      V_out      AS                  flow_mol_vap
      y_out      AS ARRAY(nocomp) OF molefraction
      Tv_out     AS                  temperature
      Pv_out     AS                  pressure
      hv_out     AS                  enth_mol_vap
#   Output 2:
      L_out1     AS                  flow_mol_liq
      x_out1     AS ARRAY(nocomp) OF molefraction
      Tl_out1    AS                  temperature
      Pl_out1    AS                  pressure
      hl_out1    AS                  enth_mol_liq
#   Output 3:
      L_out2     AS                   flow_mol_liq
      x_out2     AS ARRAY(nocomp) OF molefraction
      Tl_out2    AS                   temperature
```

```
      Pl_out2    AS                    pressure
      hl_out2    AS                    enth_mol_liq

#   Internal:
      F          AS                    flow_mol          # mixture flowrate
      H          AS                    enthflow          # enthalpy of mixture
     *P          AS                    pressure          # tray pressure
     *Q          AS                    enthflow          # tray heat load
      T          AS                    temperature       # tray temperature
      vf         AS                    vapfraction       # vapfraction
      z          AS ARRAY(nocomp) OF   molefraction      # mixture composition

  STREAM
    INPUT  1  L_in,  x_in,  Tl_in,  Pl_in,  hl_in     # liquid from tray above
    INPUT  2  F_in,  z_in,  T_in,   P_in,   h_in      # mixed phase tray feed
    OUTPUT 1  V_out, y_out, Tv_out, Pv_out, hv_out    # vapour leaving tray
    OUTPUT 2  L_out1, x_out1, Tl_out1, Pl_out1, hl_out1 # liquid overflow
    OUTPUT 3  L_out2, x_out2, Tl_out2, Pl_out2, hl_out2 # pumparound draw

  EQUATION

#   Material balance
      F     = F_in + L_in ;
      F*z   = F_in*z_in + L_in*x_in ;
      F     = V_out + L_out1 + L_out2 ;
      V_out = vf*F ;
      x_out2 = x_out1 ;

#   Energy balance
      H = F_in*h_in + L_in*hl_in + Q ;
      H = V_out*hv_out + L_out1*hl_out1 + L_out2*hl_out1 ;
      hl_out2 = hl_out1 ;

#   Temperature and Pressure Equations
      T = Tv_out = Tl_out1 = Tl_out2 ;
      P = Pv_out = Pl_out1 = Pl_out2 ;

  PROCEDURE
    ( y_out, x_out1, vf, hv_out, hl_out1 ) flash ( T, P, z ) INPUT 1
****
MODEL liq_vflow
HELP
    Calculation of the Volume Flow and Standard Liquid Volume Flow
    of a Liquid Stream.
```

```
    INPUT 1          : Stream                      type LIQUID
    OUTPUT 1         : Same Stream                 type LIQUID

    Preferred Sets   : STDTEMP (Standard Temperature, usually 60F)
                       STDPRES (Standard Pressure, usually 14.696psia)

    Result Variables : RHOL     = Liquid Density at Stream Conditions
                       VOLFL    = Liquid Volume Flow
                       STDRHOL  = Liquid Density at Std Conditions
                       STDVOLFL = Standard Liquid Volume Flow

    Parameters       : nocomp
$ENDHELP
  SET
    nocomp                          # number of components
  TYPE
#   Input:
      L_in      AS                  flow_mol_liq
      x_in      AS ARRAY(nocomp) OF molefraction
      Tl_in     AS                  temperature
      Pl_in     AS                  pressure
      hl_in     AS                  enth_mol_liq
#   Output:
      L_out     AS                  flow_mol_liq
      x_out     AS ARRAY(nocomp) OF molefraction
      Tl_out    AS                  temperature
      Pl_out    AS                  pressure
      hl_out    AS                  enth_mol_liq
#   Internal:
      VOLFl     AS                  flow_vol_liq     # vol flowrate liquid
      RHOl      AS                  dens_mol_liq     # liq molar density
      STDVOLFl  AS                  flow_vol_liq     # std vol flow liq
      STDRHOl   AS                  dens_mol_liq     # std liq mol density
     *STDTEMP   AS                  temperature      # std temperature
     *STDPRES   AS                  pressure         # std pressure
  STREAM
    INPUT 1    L_in,   x_in,   Tl_in,   Pl_in,   hl_in
    OUTPUT 1   L_out,  x_out,  Tl_out,  Pl_out,  hl_out
  EQUATION
#   Match Input to Output
      L_in = L_out ;
      x_in = x_out ;
      Tl_in = Tl_out ;
      Pl_in = Pl_out ;
      hl_in = hl_out ;
```

```
#   Calculate the Volume Flow of the Stream
      L_in = VOLFl * RHOl ;
#   Calculate the Standard Volume Flow of the STream
      L_in = STDVOLFl * STDRHOl ;
PROCEDURE
    ( RHOl )    dens_mol_liq ( Tl_in, Pl_in, x_in )   INPUT 1
    ( STDRHOl ) dens_mol_liq ( STDTEMP, STDPRES, x_in ) INPUT 1
****
MODEL vap_vflow
HELP
    Calculation of the Volume Flow, Standard Liquid Volume Flow
    and Standard Vapour Flow of a Vapour Stream.

    INPUT 1          : Stream                      type VAPOUR
    OUTPUT 1         : Same Stream                 type VAPOUR

    Preferred Sets   : STDTEMP (Standard Temperature, usually 60F)
                       STDPRES (Standard Pressure, usually 14.696psia)

    Result Variables : RHOV     = Vapour Density at Stream Conditions
                       VOLFV    = Vapour Volume Flow
                       STDRHOL  = Liquid Density at Std Conditions
                       STDVOLFL = Standard Liquid Volume Flow
                       STDVOLFV = Standard Vapour Volume Flow

    Parameters       : nocomp
$ENDHELP
  SET
    nocomp                          # number of components
  TYPE
#   Input:
      V_in      AS                  flow_mol_vap
      y_in      AS ARRAY(nocomp) OF molefraction
      Tv_in     AS                  temperature
      Pv_in     AS                  pressure
      hv_in     AS                  enth_mol_vap
#   Output:
      V_out     AS                  flow_mol_vap
      y_out     AS ARRAY(nocomp) OF molefraction
      Tv_out    AS                  temperature
      Pv_out    AS                  pressure
      hv_out    AS                  enth_mol_vap
#   Internal:
      VOLFv     AS                  flow_vol_vap     # vol flowrate vapour
      RHOv      AS                  dens_mol_vap     # vapour molar density
```

```
      STDVOLFl  AS                  flow_vol_liq      # std vol flow liq
      STDRHOl   AS                  dens_mol_liq      # std liq mol density
      STDVOLFv  AS                  flow_vol_vap      # std vol flow vap
      STDRHOv   AS                  dens_mol_vap      # std vap mol density
     *STDTEMP   AS                  temperature       # std temperature
     *STDPRES   AS                  pressure          # std pressure
  STREAM
    INPUT 1    V_in,   y_in,   Tv_in,   Pv_in,   hv_in
    OUTPUT 1   V_out,  y_out,  Tv_out,  Pv_out,  hv_out
  EQUATION
#   Match Input to Output
      V_in = V_out ;
      y_in = y_out ;
      Tv_in = Tv_out ;
      Pv_in = Pv_out ;
      hv_in = hv_out ;
#   Calculate the Volume Flow of the Stream
      V_in = VOLFv * RHOv ;
#   Calculate the Standard Liquid Volume Flow of the Stream
      V_in = STDVOLFl * STDRHOl ;
#   Calculate the Standard Vapour Volume Flow of the Stream
      V_in = STDVOLFv * STDRHOv ;
PROCEDURE
    ( RHOv )    dens_mol_vap ( Tv_in, Pv_in, y_in )     INPUT 1
    ( STDRHOl ) dens_mol_liq ( STDTEMP, STDPRES, y_in ) INPUT 1
    ( STDRHOv ) dens_mol_vap ( STDTEMP, STDPRES, y_in ) INPUT 1
****
MODEL vflow
HELP
    Calculation of the Volume Flow and Standard Liquid Volume Flow
    of a mixed Liquid-Vapour stream.

    INPUT 1          : Stream                      type MAINSTREAM
    OUTPUT 1         : Same Stream                 type MAINSTREAM

    Preferred Sets   : STDTEMP (Standard Temperature, usually 60F)
                       STDPRES (Standard Pressure, usually 14.696psia)

    Result Variables : RHOV     = Vapour Phase Density at Stream Conditions
                       VOLFV    = Vapour Phase Volume Flow
                       RHOL     = Liquid Phase Density at Stream Conditions
                       VOLFL    = Liquid Phase Volume Flow
                       VOLF     = Total Stream Volume Flow
                       STDRHOL  = Liquid Density at Std Conditions
                       STDVOLFL = Standard Liquid Volume Flow
```

```
    Parameters       : nocomp
$ENDHELP
  SET
    nocomp                          # number of components
  TYPE
#   Input:
      F_in      AS                  flow_mol
      z_in      AS ARRAY(nocomp) OF molefraction
      T_in      AS                  temperature
      P_in      AS                  pressure
      h_in      AS                  enth_mol
#   Output:
      F_out     AS                  flow_mol
      z_out     AS ARRAY(nocomp) OF molefraction
      T_out     AS                  temperature
      P_out     AS                  pressure
      h_out     AS                  enth_mol
#   Internal:
      VOLFv     AS                  flow_vol_vap     # vol flowrate vapour
      RHOv      AS                  dens_mol_vap     # vapour molar density
      VOLFl     AS                  flow_vol_liq     # vol flowrate liquid
      RHOl      AS                  dens_mol_liq     # liquid molar density
      VOLF      AS                  flow_vol         # vol flowrate total
      STDVOLFl  AS                  flow_vol_liq     # std vol flow liq
      STDRHOl   AS                  dens_mol_liq     # std liq mol density
     *STDTEMP   AS                  temperature      # std temperature
     *STDPRES   AS                  pressure         # std pressure
      x         AS ARRAY(nocomp) OF molefraction     # liquid composition
      y         AS ARRAY(nocomp) OF molefraction     # vapour composition
      vf        AS                  vapfraction      # vapour fraction
      hl        AS                  enth_mol_liq     # liq spec. enthalpy
      hv        AS                  enth_mol_vap     # vap spec. enthalpy
      V         AS                  flow_mol_vap     # vap mole flow
      L         AS                  flow_mol_liq     # liq mole flow
  STREAM
    INPUT 1    F_in,   z_in,   T_in,   P_in,   h_in
    OUTPUT 1   F_out,  z_out,  T_out,  P_out,  h_out
  EQUATION
#   Match Input to Output
      F_in = F_out ;
      z_in = z_out ;
      T_in = T_out ;
      P_in = P_out ;
      h_in = h_out ;
```

```
#   Calculate the Volume Flow of the Stream
      V = F_in * vf ;
      F_in = L + V ;
      V = VOLFv * RHOv ;
      L = VOLFl * RHOl ;
      VOLF = VOLFv + VOLFl ;
#   Calculate the Standard Volume Flow of the STream
      F_in = STDVOLFl * STDRHOl ;

PROCEDURE
    ( y, x, vf, hv, hl ) flash ( T_in, P_in, z_in ) INPUT 1
    ( RHOv   ) dens_mol_vap  ( T_in, P_in, y )      INPUT 1
    ( RHOl   ) dens_mol_liq  ( T_in, P_in, x )      INPUT 1
    ( STDRHOl ) dens_mol_liq ( STDTEMP, STDPRES, z_in ) INPUT 1
****
MODEL C5_vflow
HELP
    Calculation of the Volume Flow and Standard Liquid Volume Flow
    of a Liquid Stream and C5 COMPOSITION.

    INPUT 1          : Stream                    type LIQUID
    OUTPUT 1         : Same Stream               type LIQUID

    Preferred Sets   : STDTEMP (Standard Temperature, usually 60F)
                       STDPRES (Standard Pressure, usually 14.696psia)

    Result Variables : RHOL     = Liquid Density at Stream Conditions
                       VOLFL    = Liquid Volume Flow
                       STDRHOL  = Liquid Density at Std Conditions
                       STDVOLFL = Standard Liquid Volume Flow

    Parameters       : nocomp
$ENDHELP
  SET
    nocomp                          # number of components
  TYPE
#   Input:
      L_in      AS                  flow_mol_liq
      x_in      AS ARRAY(nocomp) OF molefraction
      Tl_in     AS                  temperature
      Pl_in     AS                  pressure
      hl_in     AS                  enth_mol_liq
#   Output:
      L_out     AS                  flow_mol_liq
      x_out     AS ARRAY(nocomp) OF molefraction
```

```
      Tl_out     AS                  temperature
      Pl_out     AS                  pressure
      hl_out     AS                  enth_mol_liq
#   Internal:
      VOLFl      AS                  flow_vol_liq     # vol flowrate liquid
      RHOl       AS                  dens_mol_liq     # liq molar density
      STDVOLFl   AS                  flow_vol_liq     # std vol flow liq
      STDRHOl    AS                  dens_mol_liq     # std liq mol density
     *STDTEMP    AS                  temperature      # std temperature
     *STDPRES    AS                  pressure         # std pressure
      C5S        AS                  molefraction     # C5's in stream
  STREAM
    INPUT 1    L_in,   x_in,   Tl_in,   Pl_in,   hl_in
    OUTPUT 1   L_out,  x_out,  Tl_out,  Pl_out,  hl_out
  EQUATION
#   Match Input to Output
      L_in = L_out ;
      x_in = x_out ;
      Tl_in = Tl_out ;
      Pl_in = Pl_out ;
      hl_in = hl_out ;
#   Calculate the Volume Flow of the Stream
      L_in = VOLFl * RHOl ;
#   Calculate the Standard Volume Flow of the STream
      L_in = STDVOLFl * STDRHOl ;
#   Calculate C5's in Stream
      C5S = X_IN(7) + X_IN(8) ;
PROCEDURE
    ( RHOl )    dens_mol_liq ( Tl_in, Pl_in, x_in )    INPUT 1
    ( STDRHOl ) dens_mol_liq ( STDTEMP, STDPRES, x_in ) INPUT 1
****
MODEL M2R
HELP
    Convert stream type from LIQUID to RVPLIQ

    INPUT 1          : Stream                    type LIQUID
    OUTPUT 1         : Same Stream               type RVPLIQ

$ENDHELP
  SET
    mainnc = 23                        # number of components
    rvpnc = 24
  TYPE
#   Input:
      F_in       AS                  flow_mol
```

```
      z_in      AS ARRAY(mainnc) OF molefraction
      T_in      AS                  temperature
      P_in      AS                  pressure
      h_in      AS                  enth_mol
#   Output:
      F_out     AS                  flow_mol
      z_out     AS ARRAY(rvpnc)  OF molefraction
      T_out     AS                  temperature
      P_out     AS                  pressure
      h_out     AS                  enth_mol
  STREAM
    INPUT 1    F_in,   z_in,   T_in,   P_in,   h_in
    OUTPUT 1   F_out,  z_out,  T_out,  P_out,  h_out
  EQUATION
#   Match Input to Output
      F_in = F_out ;
      z_in = z_out(1:mainnc) ;
      z_out(rvpnc) = 0 ;
      T_in = T_out ;
      P_in = P_out ;
      h_in = h_out ;
****
MODEL RVP
HELP
    Calculation of RVP (Reid Vapour Pressure) of a stream.

    INPUT 1        :  Stream                    type RVPLIQ
    OUTPUT 1       :  Same Stream               type RVPLIQ

    Preferred Sets :  COLDTEMP (Cold Temperature, usually 32F)
                      WARMTEMP ( Warm Temperature, usually 100F)
                      STDPRES (Standard Pressure, usually 14.696psia)
                      ZAIR (composition of Air stream)

    Result Variables : RVP      = Reid Vapour Pressure
$ENDHELP
  SET
    rvpnc = 24                          # number of components
  TYPE
#   Input:
      F_in      AS                  flow_mol_liq
      z_in      AS ARRAY(rvpnc) OF molefraction
      T_in      AS                  temperature
      P_in      AS                  pressure
      h_in      AS                  enth_mol_liq
```

```
#   Output:
      F_out      AS                    flow_mol_liq
      z_out      AS ARRAY(rvpnc) OF molefraction
      T_out      AS                    temperature
      P_out      AS                    pressure
      h_out      AS                    enth_mol_liq
#   Internal:
      RVP        AS                    pressure
     *STDPRES    AS                    pressure
     *COLDTEMP   AS                    temperature
     *WARMTEMP   AS                    temperature
     *ZAIR       AS ARRAY(rvpnc) OF molefraction
      RHO32      AS                    dens_mol_liq
      VOL32      AS                    flow_vol_liq
     *RHOA32     AS                    dens_mol_vap
      VOLA32     AS                    flow_vol_vap
      FA32       AS                    flow_mol_vap
      FMIX1      AS                    flow_mol
      ZMIX1      AS ARRAY(rvpnc) OF molefraction
      VFMIX1     AS                    vapfraction
      VOLMIX1    AS                    flow_vol
      PMIX1      AS                    pressure
      YW32       AS ARRAY(rvpnc) OF molefraction
      XST32      AS ARRAY(rvpnc) OF molefraction
      hvW32      AS                    enth_mol_vap
      hlST32     AS                    enth_mol_liq
      RHOST32    AS                    dens_mol_liq
      VOLST32    AS                    flow_vol_liq
      RHOW32     AS                    dens_mol_vap
      VOLW32     AS                    flow_vol_vap
      FSAT32     AS                    flow_mol_liq
      FW32       AS                    flow_mol_vap
     *RHOB       AS                    dens_mol_vap
      VOLB       AS                    flow_vol_vap
      FB         AS                    flow_mol_vap
      FMIX2      AS                    flow_mol
      ZMIX2      AS ARRAY(rvpnc) OF molefraction
      VFMIX2     AS                    vapfraction
      VOLMIX2    AS                    flow_vol
      PMIX2      AS                    pressure
      YVP        AS ARRAY(rvpnc) OF molefraction
      XLQ        AS ARRAY(rvpnc) OF molefraction
      hvVP       AS                    enth_mol_vap
      hlLQ       AS                    enth_mol_liq
      RHOLQ      AS                    dens_mol_liq
```

```
      VOLLQ     AS                  flow_vol_liq
      RHOVP     AS                  dens_mol_vap
      VOLVP     AS                  flow_vol_vap
      FLQ       AS                  flow_mol_liq
      FVP       AS                  flow_mol_vap
  STREAM
    INPUT 1    F_in,   z_in,   T_in,   P_in,   h_in
    OUTPUT 1   F_out,  z_out,  T_out,  P_out,  h_out
  EQUATION
#   Match Input to Output
      F_in = F_out ;
      z_in = z_out ;
      T_in = T_out ;
      P_in = P_out ;
      h_in = h_out ;
#   Calculate Air Stream volume at 32F to be 25% of RVP stream volume at 32F
      F_in = VOL32 * RHO32 ;
      VOLA32 = VOL32 * 0.25 ;
      FA32 = VOLA32 * RHOA32 ;
#   Mix Air and RVP streams at 32F
      FMIX1 = FA32 + F_in ;
      FMIX1*ZMIX1 = FA32*ZAIR + F_in*Z_in ;
      VOLMIX1 = VOLA32 + VOL32 ;
#   Constant Volume Flash of Mixed stream
      FMIX1 = FSAT32 + FW32 ;
      FW32 = FMIX1 * VFMIX1 ;
      FW32 = VOLW32 * RHOW32 ;
      FSAT32 = VOLST32 * RHOST32 ;
      VOLMIX1 = VOLST32 + VOLW32 ;
#   Mix Air and SAT streams at 32F (Air to be 400% of SAT stream)
      VOLB = VOLST32 * 4.0 ;
      FB = VOLB * RHOB ;
      FMIX2 = FB + FSAT32 ;
      FMIX2*ZMIX2 = FB*ZAIR + FSAT32*XST32 ;
      VOLMIX2 = VOLB + VOLST32 ;
#   Constant Volume Flash of Mixed stream
      FMIX2 = FLQ + FVP ;
      FVP = FMIX2 * VFMIX2 ;
      FVP = VOLVP * RHOVP ;
      FLQ = VOLLQ * RHOLQ ;
      VOLMIX2 = VOLLQ + VOLVP ;
#   Calculate Reid Vapour Pressure
      RVP = PMIX2 - STDPRES ;
PROCEDURE
    ( RHO32 )   dens_mol_liq ( COLDTEMP, STDPRES, z_in ) INPUT 1
```

```
  (YW32, XST32, VFMIX1, HVW32, HLST32) flash (COLDTEMP, PMIX1, ZMIX1) INPUT 1
    ( RHOST32 ) dens_mol_liq ( COLDTEMP, PMIX1, XST32 ) INPUT 1
    ( RHOW32 )  dens_mol_vap ( COLDTEMP, PMIX1, YW32 ) INPUT 1
  (YVP, XLQ, VFMIX2, HVVP, HLLQ) flash (WARMTEMP, PMIX2, ZMIX2) INPUT 1
    ( RHOLQ ) dens_mol_liq ( WARMTEMP, PMIX2, XLQ) INPUT 1
    ( RHOVP ) dens_mol_vap ( WARMTEMP, PMIX2, YVP ) INPUT 1
****
MODEL TBP
HELP
    Calculation of TBP (True Boiling Point) Temperature for both
    10% and 90% standard liquid volume distilled, as well as
    liquid stream volume and standard volume flows.

    INPUT 1         :  Stream                      type LIQUID
    OUTPUT 1        :  Same Stream                 type LIQUID

    Preferred Sets :  STDTEMP (Standard Temperature, usually 60F)
                      STDPRES (Standard Pressure, usually 14.696psia)

    Result Variables : RHOL     = Liquid Phase Density at Stream Conditions
                       VOLFL    = Liquid Phase Volume Flow
                       STDRHOL  = Liquid Density at Std Conditions
                       STDVOLFL = Standard Liquid Volume Flow
                       TBP10    = TBP Temperature at 10% Std Vol Distilled
                       TBP90    = TBP Temperature at 90% Std Vol Distilled

    Parameters      :  nocomp
$ENDHELP
  SET
    nocomp                          # number of components
  TYPE
#   Input:
      L_in      AS                  flow_mol_liq
      x_in      AS ARRAY(nocomp) OF molefraction
      Tl_in     AS                  temperature
      Pl_in     AS                  pressure
      hl_in     AS                  enth_mol_liq
#   Output:
      L_out     AS                  flow_mol_liq
      x_out     AS ARRAY(nocomp) OF molefraction
      Tl_out    AS                  temperature
      Pl_out    AS                  pressure
      hl_out    AS                  enth_mol_liq
#   Internal:
      VOLFl     AS                  flow_vol_liq     # vol flowrate liquid
```

```
      RHOl      AS                  dens_mol_liq     # liq molar density
      STDVOLFl  AS                  flow_vol_liq     # std vol flow liq
      STDRHOl   AS                  dens_mol_liq     # std liq mol density
      TBP10     AS                  temperature      # TBP-10% Temp
      TBP90     AS                  temperature      # TBP-90% Temp
     *STDTEMP   AS                  temperature      # std temperature
     *STDPRES   AS                  pressure         # std pressure
  STREAM
    INPUT 1    L_in,   x_in,   Tl_in,   Pl_in,   hl_in
    OUTPUT 1   L_out,  x_out,  Tl_out,  Pl_out,  hl_out
  EQUATION
#   Match Input to Output
      L_in = L_out ;
      x_in = x_out ;
      Tl_in = Tl_out ;
      Pl_in = Pl_out ;
      hl_in = hl_out ;
#   Calculate the Volume Flow of the Stream
      L_in = VOLFl * RHOl ;
#   Calculate the Standard Volume Flow of the Stream
      L_in = STDVOLFl * STDRHOl ;
PROCEDURE
    ( RHOl )    dens_mol_liq ( Tl_in, Pl_in, x_in )   INPUT 1
    ( STDRHOl ) dens_mol_liq ( STDTEMP, STDPRES, x_in ) INPUT 1
    ( TBP10, TBP90 ) TBP ( x_in, L_in )
****
PROCEDURE TBP
HELP
    Calculation of TBP (True Boiling Point) Temperature for both
    10% and 90% standard liquid volume distilled.

    INPUT       :  X = Mole Fractions of Stream
                   F = Mole Flow of Stream

    OUTPUT      :  TBP10 = TBP Temperature at 10% Std Vol Distilled
                   TBP90 = TBP Temperature at 90% Std Vol Distilled

    PARAMETERS  :  nocomp = number of compontents

    **** THIS PROCEDURE IS MEANT FOR USE WITH prop6.inp PROPERTIES ****
$ENDHELP
  INPUT
    molefraction(nocomp), flow_mol_liq
  OUTPUT
    temperature, temperature
```

```
CODE
      SUBROUTINE TBP(X, NOCOMP, F, TBP10, TBP90, IFL)
C   FORTRAN SUBROUTINE TO DETERMINE 10% AND 90% VOLUME DISTILLED TBP TERATURES
C            ------ FOR USE ONLY WITH PROP6.INP PROPERTIES -------
C
C   DECLARATIONS
C    F = MOLEFLOW, TBP10 = TBP AT 10% VOL DISTILL,
C    TBP90 = TBP AT 90% DISTILL, NOCOMP = NO COMPS,
C    SVOLF = STANDARD VOLUME FLOW OF PSUEDO-COMPS ONLY
C    IFL = SPEEDUP INTEGER FAIL FLAG, M10D = TBP10 CALCULATED,
C    M90D = TBP90 CALCULATED, X = MOLEFRACTIONS, DENS = MOLAR DENSITIES,
C    T = UPPER TBP TEMPS OF PSEUDO-COMP, VF = CUMULATIVE STD VOL FRACTIONS.
      DOUBLE PRECISION F, SVOLF, TBP10, TBP90
      INTEGER NOCOMP, IFL, M10D, M90D, I
      DOUBLE PRECISION X(23), DENS(23), T(23), VF(23)
      IFL = 3
      I = NOCOMP
C   SETUP DENSITY AND PSEUDO-COMPONENT TBP ARRAYS
      DO 100 I = 1,8
        DENS(I)= 0
        T(I) = 1
        VF(I) = 0
  100 CONTINUE
      DENS(9) = 0.469804
      DENS(10) = 0.482586
      DENS(11) = 0.445019
      DENS(12) = 0.41361
      DENS(13) = 0.386685
      DENS(14) = 0.359787
      DENS(15) = 0.333838
      DENS(16) = 0.304117
      DENS(17) = 0.276475
      DENS(18) = 0.248479
      DENS(19) = 0.217059
      DENS(20) = 0.189902
      DENS(21) = 0.168425
      DENS(22) = 0.147276
      DENS(23) = 0.127239
      T(8) = 113
      T(9) =  149
      T(10) =  185
      T(11) =  221
      T(12) =  257
      T(13) =  293
      T(14) =  329
```

```
      T(15) =  365
      T(16) =  401
      T(17) =  455
      T(18) =  509
      T(19) =  563
      T(20) =  617
      T(21) =  671
      T(22) =  743
      T(23) =  950
C   CALCULATE SVOLF
      SVOLF = (X(9) * F) / DENS(9)
      DO 105 I = 10,23
        SVOLF = SVOLF + (X(I) * F) / DENS(I)
  105 CONTINUE
C   CALCULATE CUMULATIVE STANDARD VOLUME FRACTIONS
      VF(9) = ((X(9) * F) / DENS(9)) / SVOLF
      DO 110 I = 10,23
        VF(I) = VF(I-1) + (((X(I) * F) / DENS(I)) / SVOLF)
  110 CONTINUE
C   CALCULATE TBP10 AND TBP90
      TEST = 0.0
      M10D = 0
      M90D = 0
      DO 120 I = 9,23
        IF((VF(I) .GT. 0.1) .AND. (M10D .EQ. 0)) THEN
          TBP10 = T(I-1)+(((0.1-VF(I-1))/(VF(I)-VF(I-1)))*(T(I)-T(I-1)))
          M10D = 1
        ENDIF
        IF((VF(I) .GT. 0.9) .AND. (M90D .EQ. 0)) THEN
          TBP90 = T(I-1)+(((0.9-VF(I-1))/(VF(I)-VF(I-1)))*(T(I)-T(I-1)))
          M90D = 1
        ENDIF
  120 CONTINUE
      IFL = 1
      RETURN
      END
$ENDCODE
```

E.9 report6y.speedup

The following SPEEDUP input code, `report6y.speedup`, contains the user defined reports used in both `rtodr6.speedup` and `rtopt6.speedup`.

```
REPORT C1
HELP
  Stabilizer Column (C1) Summary Profiles. Made to look similar to the
  Aspen ModelManager Output.
$ENDHELP
# Text Prcessor Setup
  ?SET strtrays=2 ?END
  ?SET strtrayf=13 ?END
  ?SET rcttrays=15 ?END
  ?SET rcttrayf=28 ?END
FIELDS
# Date and Time for report
  &1 = DATE ;
  &2 = TIME ;
  &177 = DATE ;
  &178 = TIME ;
# 4 Decimal Results: Page 1 Temperature, Pressure, Liq Flow, Liq and Vap Enth
#                    Page 2 All except Tray Vapour Fractions
  DECIMALS = 4
  NOBLANKS
# condenser/splitter
  &3 RIGHT = C1COND.T ;
  &4 RIGHT = C1COND.P ;
  &5 RIGHT = C1SPL1.HL_OUT1 ;
  &6 RIGHT = C1COND.HV_OUT ;
  &8 RIGHT = C1SPL1.L_OUT2 ;
  &179 RIGHT = C1COND.V_OUT ;
  &180 RIGHT = C1SPL1.L_OUT1 ;
  &181 RIGHT = C1COND.V_OUT ;
  &183 RIGHT = C1COND.H ;
# stripping section
?REPEAT
  &?((i*6)-3) RIGHT = "C1S.TRAY_SS(?(i))".T ;
  &?((i*6)-2) RIGHT = "C1S.TRAY_SS(?(i))".P ;
  &?((i*6)-1) RIGHT = "C1S.TRAY_SS(?(i))".HL_OUT ;
  &?((i*6)) RIGHT = "C1S.TRAY_SS(?(i))".HV_OUT ;
  &?((i*6)+2) RIGHT = "C1S.TRAY_SS(?(i))".L_OUT ;
  &?((i*3)+178) RIGHT = "C1S.TRAY_SS(?(i))".V_OUT ;
  &?((i*3)+180) RIGHT = "C1S.TRAY_SS(?(i))".H ;
?WITH i = <strtrays:strtrayf>
# feed tray
  &81 RIGHT = C1FT14.T ;
  &82 RIGHT = C1FT14.P ;
  &83 RIGHT = C1FT14.HL_OUT ;
  &84 RIGHT = C1FT14.HV_OUT ;
```

```
  &86 RIGHT = C1FT14.L_OUT ;
  &220 RIGHT = C1FT14.V_OUT ;
  &221 RIGHT = C1FT14.F_IN ;
  &223 RIGHT = C1FT14.H ;
# rectifying section
?REPEAT
  &?((i*6)-3) RIGHT = "C1R.TRAY_SS(?(i))".T ;
  &?((i*6)-2) RIGHT = "C1R.TRAY_SS(?(i))".P ;
  &?((i*6)-1) RIGHT = "C1R.TRAY_SS(?(i))".HL_OUT ;
  &?((i*6)) RIGHT = "C1R.TRAY_SS(?(i))".HV_OUT ;
  &?((i*6)+2) RIGHT = "C1R.TRAY_SS(?(i))".L_OUT ;
  &?((i*3)+179) RIGHT = "C1R.TRAY_SS(?(i))".V_OUT ;
  &?((i*3)+181) RIGHT = "C1R.TRAY_SS(?(i))".H ;
?WITH i = <rcttrays:rcttrayf>
# bottom tray
  &171 RIGHT = C1BT29.T ;
  &172 RIGHT = C1BT29.P ;
  &173 RIGHT = C1BT29.HL_OUT1 ;
  &174 RIGHT = C1BT29.HV_OUT ;
  &176 RIGHT = C1BT29.L_OUT1 + C1BT29.L_OUT2 ;
  &266 RIGHT = C1BT29.V_OUT ;
  &267 RIGHT = C1REBL.F_OUT ;
  &268 RIGHT = C1BT29.L_OUT1 + C1BT29.L_OUT2 ;
  &270 RIGHT = C1BT29.H ;
# 5 Decimal Results: Page 1 Duty
#                    Page 2 Tray Vapour Fraction
  DECIMALS = 5
  NOBLANKS
# condenser/splitter
  &7 RIGHT = C1COND.Q ;
  &182 RIGHT = C1COND.VF ;
# stripping section
?REPEAT
  &?((i*6)+1) RIGHT = "C1S.TRAY_SS(?(i))".Q ;
  &?((i*3)+179) RIGHT = "C1S.TRAY_SS(?(i))".VF ;
?WITH i = <strtrays:strtrayf>
# feed tray
  &85 RIGHT = C1FT14.Q ;
  &222 RIGHT = C1FT14.VF ;
# rectifying section
?REPEAT
  &?((i*6)+1) RIGHT = "C1R.TRAY_SS(?(i))".Q ;
  &?((i*3)+180) RIGHT = "C1R.TRAY_SS(?(i))".VF ;
?WITH i = <rcttrays:rcttrayf>
# reboiler
```

```
  &175 RIGHT = C1REBL.Q ;
  &269 RIGHT = C1BT29.VF ;
# Report Layout
DISPLAY
STABILIZER REPORT CREATED BY SPEEDUP ON %%%%%%%%%%%, %%%%%%%%%%
Page 1 of 2
                          ******************************
                          * C1 COLUMN SUMMARY PROFILES *
                          ******************************
Description: STABILIZER

STAGE TEMP        PRESSURE    LIQ ENTH      VAP ENTH      DUTY        LIQ FLOW
      F           PSI         MMBTU/LBMOL   MMBTU/LBMOL   MMBTU/HR    LBMOL/HR
   1  %%%%%%%%%% %%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%% %%%%%%%%%%
   2  %%%%%%%%%% %%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%% %%%%%%%%%%
   3  %%%%%%%%%% %%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%% %%%%%%%%%%
   4  %%%%%%%%%% %%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%% %%%%%%%%%%
   5  %%%%%%%%%% %%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%% %%%%%%%%%%
   6  %%%%%%%%%% %%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%% %%%%%%%%%%
   7  %%%%%%%%%% %%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%% %%%%%%%%%%
   8  %%%%%%%%%% %%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%% %%%%%%%%%%
   9  %%%%%%%%%% %%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%% %%%%%%%%%%
  10  %%%%%%%%%% %%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%% %%%%%%%%%%
  11  %%%%%%%%%% %%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%% %%%%%%%%%%
  12  %%%%%%%%%% %%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%% %%%%%%%%%%
  13  %%%%%%%%%% %%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%% %%%%%%%%%%
  14  %%%%%%%%%% %%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%% %%%%%%%%%%
  15  %%%%%%%%%% %%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%% %%%%%%%%%%
  16  %%%%%%%%%% %%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%% %%%%%%%%%%
  17  %%%%%%%%%% %%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%% %%%%%%%%%%
  18  %%%%%%%%%% %%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%% %%%%%%%%%%
  19  %%%%%%%%%% %%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%% %%%%%%%%%%
  20  %%%%%%%%%% %%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%% %%%%%%%%%%
  21  %%%%%%%%%% %%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%% %%%%%%%%%%
  22  %%%%%%%%%% %%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%% %%%%%%%%%%
  23  %%%%%%%%%% %%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%% %%%%%%%%%%
  24  %%%%%%%%%% %%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%% %%%%%%%%%%
  25  %%%%%%%%%% %%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%% %%%%%%%%%%
  26  %%%%%%%%%% %%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%% %%%%%%%%%%
  27  %%%%%%%%%% %%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%% %%%%%%%%%%
  28  %%%%%%%%%% %%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%% %%%%%%%%%%
  29  %%%%%%%%%% %%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%% %%%%%%%%%%

--------------------------------------------------------------------------------
STABILIZER REPORT CREATED BY SPEEDUP ON %%%%%%%%%%%, %%%%%%%%%%
```

Page 2 of 2

****
* C1 COLUMN SUMMARY PROFILES *
****

Description: STABILIZER

STAGE	VAPFLOW LBMOL/HR	FEED LBMOL/HR	LIQ PROD LBMOL/HR	VAP PROD LBMOL/HR	TRAY VFRAC MOL/MOL	TRAY ENTH MMBTU/HR
1	%%%%%%%%%%	0.0000	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%
2	%%%%%%%%%%	0.0000	0.0000	0.0000	%%%%%%%%%%	%%%%%%%%%%
3	%%%%%%%%%%	0.0000	0.0000	0.0000	%%%%%%%%%%	%%%%%%%%%%
4	%%%%%%%%%%	0.0000	0.0000	0.0000	%%%%%%%%%%	%%%%%%%%%%
5	%%%%%%%%%%	0.0000	0.0000	0.0000	%%%%%%%%%%	%%%%%%%%%%
6	%%%%%%%%%%	0.0000	0.0000	0.0000	%%%%%%%%%%	%%%%%%%%%%
7	%%%%%%%%%%	0.0000	0.0000	0.0000	%%%%%%%%%%	%%%%%%%%%%
8	%%%%%%%%%%	0.0000	0.0000	0.0000	%%%%%%%%%%	%%%%%%%%%%
9	%%%%%%%%%%	0.0000	0.0000	0.0000	%%%%%%%%%%	%%%%%%%%%%
10	%%%%%%%%%%	0.0000	0.0000	0.0000	%%%%%%%%%%	%%%%%%%%%%
11	%%%%%%%%%%	0.0000	0.0000	0.0000	%%%%%%%%%%	%%%%%%%%%%
12	%%%%%%%%%%	0.0000	0.0000	0.0000	%%%%%%%%%%	%%%%%%%%%%
13	%%%%%%%%%%	0.0000	0.0000	0.0000	%%%%%%%%%%	%%%%%%%%%%
14	%%%%%%%%%%	%%%%%%%%%%	0.0000	0.0000	%%%%%%%%%%	%%%%%%%%%%
15	%%%%%%%%%%	0.0000	0.0000	0.0000	%%%%%%%%%%	%%%%%%%%%%
16	%%%%%%%%%%	0.0000	0.0000	0.0000	%%%%%%%%%%	%%%%%%%%%%
17	%%%%%%%%%%	0.0000	0.0000	0.0000	%%%%%%%%%%	%%%%%%%%%%
18	%%%%%%%%%%	0.0000	0.0000	0.0000	%%%%%%%%%%	%%%%%%%%%%
19	%%%%%%%%%%	0.0000	0.0000	0.0000	%%%%%%%%%%	%%%%%%%%%%
20	%%%%%%%%%%	0.0000	0.0000	0.0000	%%%%%%%%%%	%%%%%%%%%%
21	%%%%%%%%%%	0.0000	0.0000	0.0000	%%%%%%%%%%	%%%%%%%%%%
22	%%%%%%%%%%	0.0000	0.0000	0.0000	%%%%%%%%%%	%%%%%%%%%%
23	%%%%%%%%%%	0.0000	0.0000	0.0000	%%%%%%%%%%	%%%%%%%%%%
24	%%%%%%%%%%	0.0000	0.0000	0.0000	%%%%%%%%%%	%%%%%%%%%%
25	%%%%%%%%%%	0.0000	0.0000	0.0000	%%%%%%%%%%	%%%%%%%%%%
26	%%%%%%%%%%	0.0000	0.0000	0.0000	%%%%%%%%%%	%%%%%%%%%%
27	%%%%%%%%%%	0.0000	0.0000	0.0000	%%%%%%%%%%	%%%%%%%%%%
28	%%%%%%%%%%	0.0000	0.0000	0.0000	%%%%%%%%%%	%%%%%%%%%%
29	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	0.0000	%%%%%%%%%%	%%%%%%%%%%

```
****
REPORT C2
HELP
  Splitter Column (C2) and Sidestripper Columns (C4, C5) Summary
  Profiles. Made to look similar to the Aspen ModelManager Output.
$ENDHELP
```

```
# Text Prcessor Setup
  ?SET strtrays=2 ?END
  ?SET strtrayf=13 ?END
FIELDS
# Date and Time for report
  &1 = DATE ;
  &2 = TIME ;
  &183 = DATE ;
  &184 = TIME ;
# 4 Decimal Results: Page 1 Temperature, Pressure, Liq Flow, Liq and Vap Enth
#                    Page 2 All except Tray Vapour Fractions
  DECIMALS = 4
  NOBLANKS
# condenser
  &3 RIGHT = C2COND.T ;
  &4 RIGHT = C2COND.P ;
  &5 RIGHT = C2COND.HL_OUT1 ;
  &6 RIGHT = 0.0000 ;
  &8 RIGHT = C2COND.L_OUT1 ;
  &185 RIGHT = 0.0000 ;
  &186 RIGHT = C2COND.L_OUT2 ;
# stripping section 1
?REPEAT
  &?((i*6)-1) RIGHT = "C2S1.TRAY_SS(?(i))".HL_OUT ;
  &?((i*6)) RIGHT = "C2S1.TRAY_SS(?(i))".HV_OUT ;
  &?((i*6)-3) RIGHT = "C2S1.TRAY_SS(?(i))".T ;
  &?((i*6)-2) RIGHT = "C2S1.TRAY_SS(?(i))".P ;
  &?((i*6)+2) RIGHT = "C2S1.TRAY_SS(?(i))".L_OUT ;
  &?((i*3)+181) RIGHT = "C2S1.TRAY_SS(?(i))".V_OUT ;
  &?((i*3)+183) RIGHT = "C2S1.TRAY_SS(?(i))".H ;
?WITH i = <strtrays:strtrayf>
# feed tray 14
  &81 RIGHT = C2FT14.T ;
  &82 RIGHT = C2FT14.P ;
  &83 RIGHT = C2FT14.HL_OUT ;
  &84 RIGHT = C2FT14.HV_OUT ;
  &86 RIGHT = C2FT14.L_OUT ;
  &223 RIGHT = C2FT14.V_OUT ;
  &224 RIGHT = C2FT14.F_IN ;
  &226 RIGHT = C2FT14.H ;
# stripping section 2
  &87 RIGHT = C2T15.T ;
  &88 RIGHT = C2T15.P ;
  &92 RIGHT = C2T15.L_OUT ;
  &227 RIGHT = C2T15.V_OUT ;
```

```
  &228 RIGHT = C2SPL1.L_OUT1 ;
  &230 RIGHT = C2T15.H ;
  &89 RIGHT = C2T15.HL_OUT ;
  &90 RIGHT = C2T15.HV_OUT ;
  &95 RIGHT = C2T16.HL_OUT ;
  &96 RIGHT = C2T16.HV_OUT ;
  &93 RIGHT = C2T16.T ;
  &94 RIGHT = C2T16.P ;
  &98 RIGHT = C2T16.L_OUT ;
  &231 RIGHT = C2T16.V_OUT ;
  &233 RIGHT = C2T16.H ;
# feed tray 17
  &99 RIGHT = C2FT17.T ;
  &100 RIGHT = C2FT17.P ;
  &101 RIGHT = C2FT17.HL_OUT ;
  &102 RIGHT = C2FT17.HV_OUT ;
  &104 RIGHT = C2FT17.L_OUT ;
  &234 RIGHT = C2FT17.V_OUT ;
  &235 RIGHT = C2FT17.F_IN ;
  &237 RIGHT = C2FT17.H ;
# stripping section 3
  &105 RIGHT = C2T18.T ;
  &106 RIGHT = C2T18.P ;
  &110 RIGHT = C2T18.L_OUT ;
  &238 RIGHT = C2T18.V_OUT ;
  &239 RIGHT = C2SPL2.L_OUT1 ;
  &241 RIGHT = C2T18.H ;
  &111 RIGHT = "C2S3.TRAY_SS(19)".T ;
  &112 RIGHT = "C2S3.TRAY_SS(19)".P ;
  &116 RIGHT = "C2S3.TRAY_SS(19)".L_OUT ;
  &242 RIGHT = "C2S3.TRAY_SS(19)".V_OUT ;
  &244 RIGHT = "C2S3.TRAY_SS(19)".H ;
  &117 RIGHT = "C2S3.TRAY_SS(20)".T ;
  &118 RIGHT = "C2S3.TRAY_SS(20)".P ;
  &122 RIGHT = "C2S3.TRAY_SS(20)".L_OUT ;
  &245 RIGHT = "C2S3.TRAY_SS(20)".V_OUT ;
  &247 RIGHT = "C2S3.TRAY_SS(20)".H ;
  &107 RIGHT = C2T18.HL_OUT ;
  &108 RIGHT = C2T18.HV_OUT ;
  &113 RIGHT = "C2S3.TRAY_SS(19)".HL_OUT ;
  &114 RIGHT = "C2S3.TRAY_SS(19)".HV_OUT ;
  &119 RIGHT = "C2S3.TRAY_SS(20)".HL_OUT ;
  &120 RIGHT = "C2S3.TRAY_SS(20)".HV_OUT ;
# feed tray 21
  &123 RIGHT = C2FT21.T ;
```

```
  &124 RIGHT = C2FT21.P ;
  &125 RIGHT = C2FT21.HL_OUT ;
  &126 RIGHT = C2FT21.HV_OUT ;
  &128 RIGHT = C2FT21.L_OUT ;
  &248 RIGHT = C2FT21.V_OUT ;
  &249 RIGHT = C2FT21.F_IN ;
  &251 RIGHT = C2FT21.H ;
# rectifying section
  &129 RIGHT = "C2R.TRAY_SS(22)".T ;
  &130 RIGHT = "C2R.TRAY_SS(22)".P ;
  &134 RIGHT = "C2R.TRAY_SS(22)".L_OUT ;
  &252 RIGHT = "C2R.TRAY_SS(22)".V_OUT ;
  &254 RIGHT = "C2R.TRAY_SS(22)".H ;
  &131 RIGHT = "C2R.TRAY_SS(22)".HL_OUT ;
  &132 RIGHT = "C2R.TRAY_SS(22)".HV_OUT ;
  &137 RIGHT = "C2R.TRAY_SS(23)".HL_OUT ;
  &138 RIGHT = "C2R.TRAY_SS(23)".HV_OUT ;
  &135 RIGHT = "C2R.TRAY_SS(23)".T ;
  &136 RIGHT = "C2R.TRAY_SS(23)".P ;
  &140 RIGHT = "C2R.TRAY_SS(23)".L_OUT ;
  &255 RIGHT = "C2R.TRAY_SS(23)".V_OUT ;
  &257 RIGHT = "C2R.TRAY_SS(23)".H ;
# bottom tray
  &141 RIGHT = C2BT24.T ;
  &142 RIGHT = C2BT24.P ;
  &143 RIGHT = C2BT24.HL_OUT1 ;
  &144 RIGHT = C2BT24.HV_OUT ;
  &146 RIGHT = C2BT24.L_OUT1 + C2BT24.L_OUT2 ;
  &258 RIGHT = C2BT24.V_OUT ;
  &259 RIGHT = C2REBL.F_OUT ;
  &260 RIGHT = C2BT24.L_OUT1 + C2BT24.L_OUT2 ;
  &262 RIGHT = C2BT24.H ;
# C4
  &147 RIGHT = "C4.TRAY_SS(1)".T ;
  &148 RIGHT = "C4.TRAY_SS(1)".P ;
  &152 RIGHT = "C4.TRAY_SS(1)".L_OUT ;
  &263 RIGHT = "C4.TRAY_SS(1)".V_OUT ;
  &264 RIGHT = "C4.TRAY_SS(1)".L_IN ;
  &265 RIGHT = "C4.TRAY_SS(1)".V_OUT ;
  &267 RIGHT = "C4.TRAY_SS(1)".H ;
  &153 RIGHT = "C4.TRAY_SS(2)".T ;
  &154 RIGHT = "C4.TRAY_SS(2)".P ;
  &158 RIGHT = "C4.TRAY_SS(2)".L_OUT ;
  &268 RIGHT = "C4.TRAY_SS(2)".V_OUT ;
  &270 RIGHT = "C4.TRAY_SS(2)".H ;
```

```
  &159 RIGHT = C4REBL.T ;
  &160 RIGHT = C4REBL.P ;
  &164 RIGHT = C4REBL.L_OUT ;
  &271 RIGHT = C4REBL.V_OUT ;
  &272 RIGHT = C4REBL.L_OUT ;
  &274 RIGHT = C4REBL.H ;
  &149 RIGHT = "C4.TRAY_SS(1)".HL_OUT ;
  &150 RIGHT = "C4.TRAY_SS(1)".HV_OUT ;
  &155 RIGHT = "C4.TRAY_SS(2)".HL_OUT ;
  &156 RIGHT = "C4.TRAY_SS(2)".HV_OUT ;
  &161 RIGHT = C4REBL.HL_OUT ;
  &162 RIGHT = C4REBL.HV_OUT ;
# C5
  &165 RIGHT = "C5.TRAY_SS(1)".T ;
  &166 RIGHT = "C5.TRAY_SS(1)".P ;
  &170 RIGHT = "C5.TRAY_SS(1)".L_OUT ;
  &275 RIGHT = "C5.TRAY_SS(1)".V_OUT ;
  &276 RIGHT = "C5.TRAY_SS(1)".L_IN ;
  &277 RIGHT = "C5.TRAY_SS(1)".V_OUT ;
  &279 RIGHT = "C5.TRAY_SS(1)".H ;
  &171 RIGHT = "C5.TRAY_SS(2)".T ;
  &172 RIGHT = "C5.TRAY_SS(2)".P ;
  &176 RIGHT = "C5.TRAY_SS(2)".L_OUT ;
  &280 RIGHT = "C5.TRAY_SS(2)".V_OUT ;
  &282 RIGHT = "C5.TRAY_SS(2)".H ;
  &177 RIGHT = C5REBL.T ;
  &178 RIGHT = C5REBL.P ;
  &182 RIGHT = C5REBL.L_OUT ;
  &283 RIGHT = C5REBL.V_OUT ;
  &284 RIGHT = C5REBL.L_OUT ;
  &286 RIGHT = C5REBL.H ;
  &167 RIGHT = "C5.TRAY_SS(1)".HL_OUT ;
  &168 RIGHT = "C5.TRAY_SS(1)".HV_OUT ;
  &173 RIGHT = "C5.TRAY_SS(2)".HL_OUT ;
  &174 RIGHT = "C5.TRAY_SS(2)".HV_OUT ;
  &179 RIGHT = C5REBL.HL_OUT ;
  &180 RIGHT = C5REBL.HV_OUT ;
# 6 Decimal Results: Page 1 Duty
#                    Page 2 Tray Vapour Fraction
  DECIMALS = 6
  NOBLANKS
# condenser
  &7 RIGHT = C2COND.Q ;
# stripping section 1
?REPEAT
```

```
  &?((i*6)+1) RIGHT = "C2S1.TRAY_SS(?(i))".Q ;
  &?((i*3)+182) RIGHT = "C2S1.TRAY_SS(?(i))".VF ;
?WITH i = <strtrays:strtrayf>
# feed tray
  &85 RIGHT = C2FT14.Q ;
  &225 RIGHT = C2FT14.VF ;
# stripping section 2
  &91 RIGHT = C2T15.Q ;
  &229 RIGHT = C2T15.VF ;
  &97 RIGHT = C2T16.Q ;
  &232 RIGHT = C2T16.VF ;
# feed tray 17
  &103 RIGHT = C2FT17.Q ;
  &236 RIGHT = C2FT17.VF ;
# stripping section 3
  &109 RIGHT = C2T18.Q ;
  &240 RIGHT = C2T18.VF ;
  &115 RIGHT = "C2S3.TRAY_SS(19)".Q ;
  &243 RIGHT = "C2S3.TRAY_SS(19)".VF ;
  &121 RIGHT = "C2S3.TRAY_SS(20)".Q ;
  &246 RIGHT = "C2S3.TRAY_SS(20)".VF ;
# feed tray 21
  &127 RIGHT = C2FT21.Q ;
  &250 RIGHT = C2FT21.VF ;
# rectifying section
  &133 RIGHT = "C2R.TRAY_SS(22)".Q ;
  &253 RIGHT = "C2R.TRAY_SS(22)".VF ;
  &139 RIGHT = "C2R.TRAY_SS(23)".Q ;
  &256 RIGHT = "C2R.TRAY_SS(23)".VF ;
# reboiler
  &145 RIGHT = C2REBL.Q ;
  &261 RIGHT = C2BT24.VF ;
# C4
  &151 RIGHT = "C4.TRAY_SS(1)".Q ;
  &266 RIGHT = "C4.TRAY_SS(1)".VF ;
  &157 RIGHT = "C4.TRAY_SS(2)".Q ;
  &269 RIGHT = "C4.TRAY_SS(2)".VF ;
  &163 RIGHT = C4REBL.Q ;
  &273 RIGHT = C4REBL.VF ;
# C5
  &169 RIGHT = "C5.TRAY_SS(1)".Q ;
  &278 RIGHT = "C5.TRAY_SS(1)".VF ;
  &175 RIGHT = "C5.TRAY_SS(2)".Q ;
  &281 RIGHT = "C5.TRAY_SS(2)".VF ;
  &181 RIGHT = C5REBL.Q ;
```

```
  &285 RIGHT = C5REBL.VF ;
# Report Layout
DISPLAY
SPLITTER REPORT CREATED BY SPEEDUP ON %%%%%%%%%%%, %%%%%%%%%%
Page 1 of 2
                          ******************************
                          * C2 COLUMN SUMMARY PROFILES *
                          ******************************
Description: SPLITTER

STAGE TEMP        PRESSURE   LIQ ENTH      VAP ENTH      DUTY       LIQ FLOW
      F           PSI        MMBTU/LBMOL   MMBTU/LBMOL   MMBTU/HR   LBMOL/HR
    1 %%%%%%%%%% %%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%% %%%%%%%%%%
    2 %%%%%%%%%% %%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%% %%%%%%%%%%
    3 %%%%%%%%%% %%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%% %%%%%%%%%%
    4 %%%%%%%%%% %%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%% %%%%%%%%%%
    5 %%%%%%%%%% %%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%% %%%%%%%%%%
    6 %%%%%%%%%% %%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%% %%%%%%%%%%
    7 %%%%%%%%%% %%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%% %%%%%%%%%%
    8 %%%%%%%%%% %%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%% %%%%%%%%%%
    9 %%%%%%%%%% %%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%% %%%%%%%%%%
   10 %%%%%%%%%% %%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%% %%%%%%%%%%
   11 %%%%%%%%%% %%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%% %%%%%%%%%%
   12 %%%%%%%%%% %%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%% %%%%%%%%%%
   13 %%%%%%%%%% %%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%% %%%%%%%%%%
   14 %%%%%%%%%% %%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%% %%%%%%%%%%
   15 %%%%%%%%%% %%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%% %%%%%%%%%%
   16 %%%%%%%%%% %%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%% %%%%%%%%%%
   17 %%%%%%%%%% %%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%% %%%%%%%%%%
   18 %%%%%%%%%% %%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%% %%%%%%%%%%
   19 %%%%%%%%%% %%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%% %%%%%%%%%%
   20 %%%%%%%%%% %%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%% %%%%%%%%%%
   21 %%%%%%%%%% %%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%% %%%%%%%%%%
   22 %%%%%%%%%% %%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%% %%%%%%%%%%
   23 %%%%%%%%%% %%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%% %%%%%%%%%%
   24 %%%%%%%%%% %%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%% %%%%%%%%%%

                          ******************************
                          * C4 COLUMN SUMMARY PROFILES *
                          ******************************
Description: HI-FD SIDESTRIPPER

STAGE TEMP        PRESSURE   LIQ ENTH      VAP ENTH      DUTY       LIQ FLOW
      F           PSI        MMBTU/LBMOL   MMBTU/LBMOL   MMBTU/HR   LBMOL/HR
    1 %%%%%%%%%% %%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%% %%%%%%%%%%
```

2	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%%	%%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%
3	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%%	%%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%

****
* C5 COLUMN SUMMARY PROFILES *
****

Description: LO-FD SIDESTRIPPER

STAGE	TEMP F	PRESSURE PSI	LIQ ENTH MMBTU/LBMOL	VAP ENTH MMBTU/LBMOL	DUTY MMBTU/HR	LIQ FLOW LBMOL/HR
1	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%%	%%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%
2	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%%	%%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%
3	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%%	%%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%

SPLITTER REPORT CREATED BY SPEEDUP ON %%%%%%%%%%, %%%%%%%%%%

****
* C2 COLUMN SUMMARY PROFILES *
****

Description: SPLITTER

STAGE	VAPFLOW LBMOL/HR	FEED LBMOL/HR	LIQ PROD LBMOL/HR	VAP PROD LBMOL/HR	TRAY VFRAC MOL/MOL	TRAY ENTH MMBTU/HR
1	%%%%%%%%%%	0.0000	%%%%%%%%%%	0.0000	----------	----------
2	%%%%%%%%%%	0.0000	0.0000	0.0000	%%%%%%%%%%	%%%%%%%%%%
3	%%%%%%%%%%	0.0000	0.0000	0.0000	%%%%%%%%%%	%%%%%%%%%%
4	%%%%%%%%%%	0.0000	0.0000	0.0000	%%%%%%%%%%	%%%%%%%%%%
5	%%%%%%%%%%	0.0000	0.0000	0.0000	%%%%%%%%%%	%%%%%%%%%%
6	%%%%%%%%%%	0.0000	0.0000	0.0000	%%%%%%%%%%	%%%%%%%%%%
7	%%%%%%%%%%	0.0000	0.0000	0.0000	%%%%%%%%%%	%%%%%%%%%%
8	%%%%%%%%%%	0.0000	0.0000	0.0000	%%%%%%%%%%	%%%%%%%%%%
9	%%%%%%%%%%	0.0000	0.0000	0.0000	%%%%%%%%%%	%%%%%%%%%%
10	%%%%%%%%%%	0.0000	0.0000	0.0000	%%%%%%%%%%	%%%%%%%%%%
11	%%%%%%%%%%	0.0000	0.0000	0.0000	%%%%%%%%%%	%%%%%%%%%%
12	%%%%%%%%%%	0.0000	0.0000	0.0000	%%%%%%%%%%	%%%%%%%%%%
13	%%%%%%%%%%	0.0000	0.0000	0.0000	%%%%%%%%%%	%%%%%%%%%%
14	%%%%%%%%%%	%%%%%%%%%%	0.0000	0.0000	%%%%%%%%%%	%%%%%%%%%%
15	%%%%%%%%%%	0.0000	%%%%%%%%%%	0.0000	%%%%%%%%%%	%%%%%%%%%%
16	%%%%%%%%%%	0.0000	0.0000	0.0000	%%%%%%%%%%	%%%%%%%%%%
17	%%%%%%%%%%	%%%%%%%%%%	0.0000	0.0000	%%%%%%%%%%	%%%%%%%%%%
18	%%%%%%%%%%	0.0000	%%%%%%%%%%	0.0000	%%%%%%%%%%	%%%%%%%%%%
19	%%%%%%%%%%	0.0000	0.0000	0.0000	%%%%%%%%%%	%%%%%%%%%%
20	%%%%%%%%%%	0.0000	0.0000	0.0000	%%%%%%%%%%	%%%%%%%%%%
21	%%%%%%%%%%	%%%%%%%%%%	0.0000	0.0000	%%%%%%%%%%	%%%%%%%%%%

```
  22  %%%%%%%%%%      0.0000      0.0000      0.0000 %%%%%%%%%% %%%%%%%%%%
  23  %%%%%%%%%%      0.0000      0.0000      0.0000 %%%%%%%%%% %%%%%%%%%%
  24  %%%%%%%%%% %%%%%%%%%% %%%%%%%%%%      0.0000 %%%%%%%%%% %%%%%%%%%%

                        ******************************
                        * C4 COLUMN SUMMARY PROFILES *
                        ******************************
Description: HI-FD SIDESTRIPPER

STAGE VAPFLOW     FEED        LIQ PROD    VAP PROD    TRAY VFRAC TRAY ENTH
      LBMOL/HR    LBMOL/HR    LBMOL/HR    LBMOL/HR    MOL/MOL    MMBTU/HR
   1  %%%%%%%%%% %%%%%%%%%%      0.0000 %%%%%%%%%% %%%%%%%%%% %%%%%%%%%%
   2  %%%%%%%%%%      0.0000      0.0000      0.0000 %%%%%%%%%% %%%%%%%%%%
   3  %%%%%%%%%%      0.0000 %%%%%%%%%%      0.0000 %%%%%%%%%% %%%%%%%%%%

                        ******************************
                        * C5 COLUMN SUMMARY PROFILES *
                        ******************************
Description: LO-FD SIDESTRIPPER

STAGE VAPFLOW     FEED        LIQ PROD    VAP PROD    TRAY VFRAC TRAY ENTH
      LBMOL/HR    LBMOL/HR    LBMOL/HR    LBMOL/HR    MOL/MOL    MMBTU/HR
   1  %%%%%%%%%% %%%%%%%%%%      0.0000 %%%%%%%%%% %%%%%%%%%% %%%%%%%%%%
   2  %%%%%%%%%%      0.0000      0.0000      0.0000 %%%%%%%%%% %%%%%%%%%%
   3  %%%%%%%%%%      0.0000 %%%%%%%%%%      0.0000 %%%%%%%%%% %%%%%%%%%%

--------------------------------------------------------------------------------
****
REPORT STREAMSUM
HELP
  Stream Summary for Stabilizer-Splitter Flowsheet (C1, E1, C2,
  C4, C5). Made to look similar to the Aspen ModelManager Output.
$ENDHELP
# Text Prcessor Setup
  ?SET nocomp=23 ?END
FIELDS
# Date and Time for report
  &1 = DATE ;
  &2 = TIME ;
  &167 = DATE ;
  &168 = TIME ;
  &333 = DATE ;
  &334 = TIME ;
# Set format for report
  DECIMALS = 2
```

```
  NOBLANKS
# Page 1 Stream Component Mole Fractions
?REPEAT
  &?((i*5)-2) RIGHT = C1SPL1.X_OUT1(?(i)) ;
  &?((i*5)-1) RIGHT = C4REBL.X_OUT(?(i)) ;
  &?(i*5) RIGHT = F41.X_OUT(?(i)) ;
  &?((i*5)+1) RIGHT = F51.X_OUT(?(i)) ;
  &?((i*5)+2) RIGHT = C5REBL.X_OUT(?(i)) ;
?WITH i = <1:nocomp>
# Page 1 Stream Mole Flows
  &118 RIGHT = C1SPL1.L_OUT1 ;
  &119 RIGHT = C4REBL.L_OUT ;
  &120 RIGHT = F41.L_OUT ;
  &121 RIGHT = F51.L_OUT ;
  &122 RIGHT = C5REBL.L_OUT ;
# Page 1 Stream Volume Flows (cuft/hr)
  &123 RIGHT = F13.VOLFL ;
  &124 RIGHT = F42.VOLFL ;
  &125 RIGHT = F41.VOLFL ;
  &126 RIGHT = F51.VOLFL ;
  &127 RIGHT = F52.VOLFL ;
# Page 1 Stream Volume Flows (bbl/day)
  &128 RIGHT = F13.VOLFL / 0.23394 ;
  &129 RIGHT = F42.VOLFL / 0.23394 ;
  &130 RIGHT = F41.VOLFL / 0.23394 ;
  &131 RIGHT = F51.VOLFL / 0.23394 ;
  &132 RIGHT = F52.VOLFL / 0.23394 ;
# Page 1 Stream Standard Liquid Volume Flow (cuft/hr)
  &133 RIGHT = F13.STDVOLFL ;
  &134 RIGHT = F42.STDVOLFL ;
  &135 RIGHT = F41.STDVOLFL ;
  &136 RIGHT = F51.STDVOLFL ;
  &137 RIGHT = F52.STDVOLFL ;
# Page 1 Stream Standard Liquid Volume Flow (bbl/day)
  &138 RIGHT = F13.STDVOLFL / 0.23394 ;
  &139 RIGHT = F42.STDVOLFL / 0.23394 ;
  &140 RIGHT = F41.STDVOLFL / 0.23394 ;
  &141 RIGHT = F51.STDVOLFL / 0.23394 ;
  &142 RIGHT = F52.STDVOLFL / 0.23394 ;
# Page 1 Stream Temperatures
  &143 RIGHT = C1SPL1.TL_OUT1 ;
  &144 RIGHT = C4REBL.TL_OUT ;
  &145 RIGHT = F41.TL_OUT ;
  &146 RIGHT = F51.TL_OUT ;
  &147 RIGHT = C5REBL.TL_OUT ;
```

```
# Page 1 Stream Pressures
  &148 RIGHT = C1SPL1.PL_OUT1 ;
  &149 RIGHT = C4REBL.PL_OUT ;
  &150 RIGHT = F41.PL_OUT ;
  &151 RIGHT = F51.PL_OUT ;
  &152 RIGHT = C5REBL.PL_OUT ;
# Page 1 Stream Enthalpies
  &153 RIGHT = C1SPL1.HL_OUT1 ;
  &154 RIGHT = C4REBL.HL_OUT ;
  &155 RIGHT = F41.HL_OUT ;
  &156 RIGHT = F51.HL_OUT ;
  &157 RIGHT = C5REBL.HL_OUT ;
# Page 1 Stream Densities
  &158 RIGHT = F13.RHOL ;
  &159 RIGHT = F42.RHOL ;
  &160 RIGHT = F41.RHOL ;
  &161 RIGHT = F51.RHOL ;
  &162 RIGHT = F52.RHOL ;
# Page 1 TBP10 TemperAtures
  &163 RIGHT = F42.TBP10 ;
  &164 RIGHT = F52.TBP10 ;
# Page 1 TBP90 Temperatures
  &165 RIGHT = F42.TBP90 ;
  &166 RIGHT = F52.TBP90 ;
# Page 2 Stream Component Mole Fractions
?REPEAT
  &?((i*5)+164) RIGHT = C2COND.X_OUT2(?(i)) ;
  &?((i*5)+165) RIGHT = E1.Z_OUT(?(i)) ;
  &?((i*5)+166) RIGHT = F22A.X_OUT(?(i)) ;
  &?((i*5)+167) RIGHT = F62.X_OUT(?(i)) ;
  &?((i*5)+168) RIGHT = MFEED.Z_OUT(?(i)) ;
?WITH i = <1:nocomp>
# Page 2 Stream Mole Flows
  &284 RIGHT = C2COND.L_OUT2 ;
  &285 RIGHT = E1.F_OUT ;
  &286 RIGHT = F22A.L_OUT ;
  &287 RIGHT = F62.L_OUT ;
  &288 RIGHT = MFEED.F_OUT ;
# Page 2 Stream Volume Flows (cuft/hr)
  &289 RIGHT = F31.VOLFL ;
  &290 RIGHT = F22.VOLF ;
  &291 RIGHT = F22A.VOLFL ;
  &292 RIGHT = F62.VOLFL ;
  &293 RIGHT = F01.VOLF ;
# Page 2 Stream Volume Flows (bbl/day)
```

```
  &294 RIGHT = F31.VOLFL / 0.23394 ;
  &295 RIGHT = F22.VOLF / 0.23394 ;
  &296 RIGHT = F22A.VOLFL / 0.23394 ;
  &297 RIGHT = F62.VOLFL / 0.23394 ;
  &298 RIGHT = F01.VOLF / 0.23394 ;
# Page 2 Stream Standard Liquid Volume Flows (cuft/hr)
  &299 RIGHT = F31.STDVOLFL ;
  &300 RIGHT = F22.STDVOLFL ;
  &301 RIGHT = F22A.STDVOLFL ;
  &302 RIGHT = F62.STDVOLFL ;
  &303 RIGHT = F01.STDVOLFL ;
# Page 2 Stream Standard Liquid Volume Flows (bbl/day)
  &304 RIGHT = F31.STDVOLFL / 0.23394 ;
  &305 RIGHT = F22.STDVOLFL / 0.23394 ;
  &306 RIGHT = F22A.STDVOLFL / 0.23394 ;
  &307 RIGHT = F62.STDVOLFL / 0.23394 ;
  &308 RIGHT = F01.STDVOLFL / 0.23394 ;
# Page 2 Stream Temperatures
  &309 RIGHT = C2COND.TL_OUT2 ;
  &310 RIGHT = E1.T_OUT ;
  &311 RIGHT = F22A.TL_OUT ;
  &312 RIGHT = F62.TL_OUT ;
  &313 RIGHT = MFEED.T_OUT ;
# Page 2 Stream Pressures
  &314 RIGHT = C2COND.PL_OUT2 ;
  &315 RIGHT = E1.P_OUT ;
  &316 RIGHT = F22A.PL_OUT ;
  &317 RIGHT = F62.PL_OUT ;
  &318 RIGHT = MFEED.P_OUT ;
# Page 2 Vapour Fractions
  &319 RIGHT = F22.VF ;
  &320 RIGHT = F01.VF ;
# Page 2 Stream Enthalpies
  &321 RIGHT = C2COND.HL_OUT2 ;
  &322 RIGHT = E1.H_OUT ;
  &323 RIGHT = F22A.HL_OUT ;
  &324 RIGHT = F62.HL_OUT ;
  &325 RIGHT = MFEED.H_OUT ;
# Page 2 Stream Densities
  &326 RIGHT = F31.RHOL ;
  &327 RIGHT = F22.F_OUT / F22.VOLF ;
  &328 RIGHT = F22A.RHOL ;
  &329 RIGHT = F62.RHOL ;
  &330 RIGHT = F01.F_OUT / F01.VOLF ;
# Page 2 Reid Vapour Pressures
```

```
  &331 RIGHT = RVP31.RVP ;
# Page 2 Bubble Point Temps
  &332 RIGHT = C2COND.T_BUBBLE ;
# Page 3 Stream Mole Fractions
?REPEAT
  &?((i*5)+330) RIGHT = C1COND.Y_OUT(?(i)) ;
  &?((i*5)+331) RIGHT = F21.Z_OUT(?(i)) ;
  &?((i*5)+332) RIGHT = F12.X_OUT(?(i)) ;
  &?((i*5)+333) RIGHT = F61.Z_OUT(?(i)) ;
  &?((i*5)+334) RIGHT = F32.X_OUT(?(i)) ;
?WITH i = <1:nocomp>
# Page 3 Stream Component Mole Flows
  &450 RIGHT = C1COND.V_OUT ;
  &451 RIGHT = F21.F_OUT ;
  &452 RIGHT = F12.L_OUT ;
  &453 RIGHT = F61.F_OUT ;
  &454 RIGHT = F32.L_OUT ;
# Page 3 Stream Volume Flows (cuft/hr)
  &455 RIGHT = F11.VOLFV ;
  &456 RIGHT = F21.VOLF ;
  &457 RIGHT = F12.VOLFL ;
  &458 RIGHT = F61.VOLF ;
  &459 RIGHT = F32.VOLFL ;
# Page 3 Stream Volume Flows (bbl/day)
  &460 RIGHT = F11.VOLFV / 0.23394 ;
  &461 RIGHT = F21.VOLF / 0.23394 ;
  &462 RIGHT = F12.VOLFL / 0.23394 ;
  &463 RIGHT = F61.VOLF / 0.23394 ;
  &464 RIGHT = F32.VOLFL / 0.23394 ;
# Page 3 Stream Standard Liquid Volume Flows (cuft/hr)
  &465 RIGHT = F11.STDVOLFL ;
  &466 RIGHT = F21.STDVOLFL ;
  &467 RIGHT = F12.STDVOLFL ;
  &468 RIGHT = F61.STDVOLFL ;
  &469 RIGHT = F32.STDVOLFL ;
# Page 3 Stream Standard Liquid Volume Flows (bbl/day)
  &470 RIGHT = F11.STDVOLFL / 0.23394 ;
  &471 RIGHT = F21.STDVOLFL / 0.23394 ;
  &472 RIGHT = F12.STDVOLFL / 0.23394 ;
  &473 RIGHT = F61.STDVOLFL / 0.23394 ;
  &474 RIGHT = F32.STDVOLFL / 0.23394 ;
# Page 3 Stream Temperatures
  &475 RIGHT = C1COND.TV_OUT ;
  &476 RIGHT = F21.T_OUT ;
  &477 RIGHT = F12.TL_OUT ;
```

```
  &478 RIGHT = F61.T_OUT ;
  &479 RIGHT = F32.TL_OUT ;
# Page 3 Stream Pressures
  &480 RIGHT = C1COND.PV_OUT ;
  &481 RIGHT = F21.P_OUT ;
  &482 RIGHT = F12.PL_OUT ;
  &483 RIGHT = F61.P_OUT ;
  &484 RIGHT = F32.PL_OUT ;
# Page 3 Vapour Fractions
  &485 RIGHT = F21 VF ;
  &486 RIGHT = F61.VF ;
# Page 3 Stream Enthalpies
  &487 RIGHT = C1COND.HV_OUT ;
  &488 RIGHT = F21.H_OUT ;
  &489 RIGHT = F12.HL_OUT ;
  &490 RIGHT = F61.H_OUT ;
  &491 RIGHT = F32.HL_OUT ;
# Page 3 Stream Densities
  &492 RIGHT = F11.RHOV ;
  &493 RIGHT = F21.F_OUT / F21.VOLF ;
  &494 RIGHT = F12.RHOL ;
  &495 RIGHT = F61.F_OUT / F61.VOLF ;
  &496 RIGHT = F32.RHOL ;
# Page 3 Standard Volume Flows
  &497 RIGHT = F11.STDVOLFV ;
# Report Layout
DISPLAY
STABILIZER-SPLITTER REPORT CREATED BY SPEEDUP ON %%%%%%%%%%%, %%%%%%%%%%
Page 1 of 3
                              ******************
                              * STREAM SUMMARY *
                              ******************

STREAM NAME:            DIST-C4    HV-ISO     IC1        IC2        LT-DIST
                From    C1         C4         C2         C2         C5
                To                            C4         C5
                Phase   LIQUID     LIQUID     LIQUID     LIQUID     LIQUID
COMPONENT MOLE FRACTIONS
  H2                    %%%%%%%%%% %%%%%%%%%% %%%%%%%%%% %%%%%%%%%% %%%%%%%%%%
  METHANE               %%%%%%%%%% %%%%%%%%%% %%%%%%%%%% %%%%%%%%%% %%%%%%%%%%
  ETHANE                %%%%%%%%%% %%%%%%%%%% %%%%%%%%%% %%%%%%%%%% %%%%%%%%%%
  PROPANE               %%%%%%%%%% %%%%%%%%%% %%%%%%%%%% %%%%%%%%%% %%%%%%%%%%
  IBUTANE               %%%%%%%%%% %%%%%%%%%% %%%%%%%%%% %%%%%%%%%% %%%%%%%%%%
  BUTANE                %%%%%%%%%% %%%%%%%%%% %%%%%%%%%% %%%%%%%%%% %%%%%%%%%%
  IPENTANE              %%%%%%%%%% %%%%%%%%%% %%%%%%%%%% %%%%%%%%%% %%%%%%%%%%
```

PENTANE	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%
BP45T65	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%
BP65T85	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%
BP85T105	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%
BP105T125	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%
BP125T145	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%
BP145T165	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%
BP165T185	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%
BP185T205	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%
BP205T235	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%
BP235T265	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%
BP265T295	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%
BP295T325	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%
BP325T355	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%
BP355T395	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%
BP395T510	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%
MOLE FLOW [LBMOL/HR]	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%
VOLUME FLOW [CUFT/HR]	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%
VOLUME FLOW [BBL/DAY]	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%
STD VOL FLOW [CUFT/HR]	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%
STD VOL FLOW [BBL/DAY]	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%
TEMPERATURE [F]	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%
PRESSURE [PSI]	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%
VAPOUR FRACTION	0.0	0.0	0.0	0.0	0.0
ENTHALPY [MMBTU/LBMOL]	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%
DENSITY [LBMOL/CUFT]	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%
TBP TEMP, 0.1 VOL [F]		%%%%%%%%%%			%%%%%%%%%%
TBP TEMP, 0.9 VOL [F]		%%%%%%%%%%			%%%%%%%%%%
REID VAP PRESS [PSI]					
BUBBLE POINT TEMP [F]					
STD VAP FLOW [CUFT/HR]					

STABILIZER-SPLITTER REPORT CREATED BY SPEEDUP ON %%%%%%%%%%, %%%%%%%%%%
Page 2 of 3

****
* STREAM SUMMARY *
****

STREAM NAME:		LT-ISO	SPLT	SPLTFEED	SSFEED	STAB
	From	C2	E1	C1	C2	MFEED
	To		C2	E1		C1
	Phase	LIQUID	MIXED	LIQUID	LIQUID	MIXED
COMPONENT FLOWS [MOLE FRACTION]						
H2		%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%
METHANE		%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%

ETHANE	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%
PROPANE	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%
IBUTANE	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%
BUTANE	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%
IPENTANE	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%
PENTANE	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%
BP45T65	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%
BP65T85	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%
BP85T105	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%
BP105T125	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%
BP125T145	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%
BP145T165	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%
BP165T185	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%
BP185T205	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%
BP205T235	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%
BP235T265	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%
BP265T295	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%
BP295T325	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%
BP325T355	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%
BP355T395	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%
BP395T510	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%
MOLE FLOW [LBMOL/HR]	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%
VOLUME FLOW [CUFT/HR]	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%
VOLUME FLOW [BBL/DAY]	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%
STD VOL FLOW [CUFT/HR]	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%
STD VOL FLOW [BBL/DAY]	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%
TEMPERATURE [F]	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%
PRESSURE [PSI]	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%
VAPOUR FRACTION	0.0	%%%%%%%%%%	0.0	0.0	%%%%%%%%%%
ENTHALPY [MMBTU/LBMOL]	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%
DENSITY [LBMOL/CUFT]	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%
TBP TEMP, 0.1 VOL [F]					
TBP TEMP, 0.9 VOL [F]					
REID VAP PRESS [PSI]	%%%%%%%%%%				
BUBBLE POINT TEMP [F]	%%%%%%%%%%				
STD VAP FLOW [CUFT/HR]					

STABILIZER-SPLITTER REPORT CREATED BY SPEEDUP ON %%%%%%%%%%, %%%%%%%%%%
Page 3 of 3

****
* STREAM SUMMARY *
****

STREAM NAME:	VAP	C1REBL	C1RFLX	C2REBL	C2RFLX
From	C1				

To Phase	VAPOUR	MIXED	LIQUID	MIXED	LIQUID
COMPONENT FLOWS [MOLE FRACTION]					
H2	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%
METHANE	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%
ETHANE	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%
PROPANE	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%
IBUTANE	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%
BUTANE	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%
IPENTANE	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%
PENTANE	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%
BP45T65	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%
BP65T85	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%
BP85T105	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%
BP105T125	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%
BP125T145	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%
BP145T165	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%
BP165T185	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%
BP185T205	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%
BP205T235	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%
BP235T265	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%
BP265T295	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%
BP295T325	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%
BP325T355	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%
BP355T395	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%
BP395T510	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%
MOLE FLOW [LBMOL/HR]	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%
VOLUME FLOW [CUFT/HR]	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%
VOLUME FLOW [BBL/DAY]	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%
STD VOL FLOW [CUFT/HR]	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%
STD VOL FLOW [BBL/DAY]	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%
TEMPERATURE [F]	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%
PRESSURE [PSI]	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%
VAPOUR FRACTION	1.0	%%%%%%%%%%	0.0	%%%%%%%%%%	0.0
ENTHALPY [MMBTU/LBMOL]	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%
DENSITY [LBMOL/CUFT]	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%	%%%%%%%%%%
TBP TEMP, 0.1 VOL [F]					
TBP TEMP, 0.9 VOL [F]					
REID VAP PRESS [PSI]					
BUBBLE POINT TEMP [F]					
STD VAP FLOW [CUFT/HR]	%%%%%%%%%%				

```
****
REPORT TEAR
HELP
```

```
  Tear Stream Report for the return streams from and Sidestripper Columns
  (C4, C5) to the Splitter Column (C2).
$ENDHELP
# Text Prcessor Setup
  ?SET nocomp=23 ?END
FIELDS
# Date and Time for report
  &1 = DATE ;
  &2 = TIME ;
  &84 = DATE ;
  &85 = TIME ;
  &167 = DATE ;
  &168 = TIME ;
  &250 = DATE ;
  &251 = TIME ;
  DECIMALS = 4
  NOBLANKS
?REPEAT
  &?(i*3) RIGHT = C1TR1.Z_IN(?(i)) ;
  &?((i*3)+1) RIGHT = C1TR1.Z_OUT(?(i)) ;
  &?((i*3)+2) RIGHT = C1TR1.Z_SLACK(?(i)) ;
  &?((i*3)+83) RIGHT = C2TR1.Z_IN(?(i)) ;
  &?((i*3)+84) RIGHT = C2TR1.Z_OUT(?(i)) ;
  &?((i*3)+85) RIGHT = C2TR1.Z_SLACK(?(i)) ;
  &?((i*3)+166) RIGHT = C2TR2.Z_IN(?(i)) ;
  &?((i*3)+167) RIGHT = C2TR2.Z_OUT(?(i)) ;
  &?((i*3)+168) RIGHT = C2TR2.Z_SLACK(?(i)) ;
  &?((i*3)+249) RIGHT = C2TR3.Z_IN(?(i)) ;
  &?((i*3)+250) RIGHT = C2TR3.Z_OUT(?(i)) ;
  &?((i*3)+251) RIGHT = C2TR3.Z_SLACK(?(i)) ;
?WITH i = <1:nocomp>
  &72 RIGHT = C1TR1.F_IN ;
  &73 RIGHT = C1TR1.F_OUT ;
  &74 RIGHT = C1TR1.F_SLACK ;
  &75 RIGHT = C1TR1.T_IN ;
  &76 RIGHT = C1TR1.T_OUT ;
  &77 RIGHT = C1TR1.T_SLACK ;
  &78 RIGHT = C1TR1.P_IN ;
  &79 RIGHT = C1TR1.P_OUT ;
  &80 RIGHT = C1TR1.P_SLACK ;
  &81 RIGHT = C1TR1.H_IN ;
  &82 RIGHT = C1TR1.H_OUT ;
  &83 RIGHT = C1TR1.H_SLACK ;
  &155 RIGHT = C2TR1.F_IN ;
  &156 RIGHT = C2TR1.F_OUT ;
```

```
  &157 RIGHT = C2TR1.F_SLACK ;
  &158 RIGHT = C2TR1.T_IN ;
  &159 RIGHT = C2TR1.T_OUT ;
  &160 RIGHT = C2TR1.T_SLACK ;
  &161 RIGHT = C2TR1.P_IN ;
  &162 RIGHT = C2TR1.P_OUT ;
  &163 RIGHT = C2TR1.P_SLACK ;
  &164 RIGHT = C2TR1.H_IN ;
  &165 RIGHT = C2TR1.H_OUT ;
  &166 RIGHT = C2TR1.H_SLACK ;
  &238 RIGHT = C2TR2.F_IN ;
  &239 RIGHT = C2TR2.F_OUT ;
  &240 RIGHT = C2TR2.F_SLACK ;
  &241 RIGHT = C2TR2.T_IN ;
  &242 RIGHT = C2TR2.T_OUT ;
  &243 RIGHT = C2TR2.T_SLACK ;
  &244 RIGHT = C2TR2.P_IN ;
  &245 RIGHT = C2TR2.P_OUT ;
  &246 RIGHT = C2TR2.P_SLACK ;
  &247 RIGHT = C2TR2.H_IN ;
  &248 RIGHT = C2TR2.H_OUT ;
  &249 RIGHT = C2TR2.H_SLACK ;
  &331 RIGHT = C2TR3.F_IN ;
  &332 RIGHT = C2TR3.F_OUT ;
  &333 RIGHT = C2TR3.F_SLACK ;
  &334 RIGHT = C2TR3.T_IN ;
  &335 RIGHT = C2TR3.T_OUT ;
  &336 RIGHT = C2TR3.T_SLACK ;
  &337 RIGHT = C2TR3.P_IN ;
  &338 RIGHT = C2TR3.P_OUT ;
  &339 RIGHT = C2TR3.P_SLACK ;
  &340 RIGHT = C2TR3.H_IN ;
  &341 RIGHT = C2TR3.H_OUT ;
  &342 RIGHT = C2TR3.H_SLACK ;
DISPLAY
SPLITTER TEAR STREAM REPORT CREATED BY SPEEDUP ON %%%%%%%%%%%, %%%%%%%%%%
Page 1 of 2
                    *********************************
                    * C1REBL -> C1BT29 TEAR STREAM *
                    *********************************

STREAM NAME:            IN             OUT            SLACK
COMPONENT FLOWS [MOLE FRACTION]
  H2                   %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%%
  METHANE              %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%%
```

```
  ETHANE                %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%%
  PROPANE               %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%%
  IBUTANE               %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%%
  BUTANE                %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%%
  IPENTANE              %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%%
  PENTANE               %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%%
  BP45T65               %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%%
  BP65T85               %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%%
  BP85T105              %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%%
  BP105T125             %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%%
  BP125T145             %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%%
  BP145T165             %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%%
  BP165T185             %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%%
  BP185T205             %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%%
  BP205T235             %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%%
  BP235T265             %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%%
  BP265T295             %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%%
  BP295T325             %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%%
  BP325T355             %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%%
  BP355T395             %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%%
  BP395T510             %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%%
MOLE FLOW [LBMOL/HR]    %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%%
TEMPERATURE [F]         %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%%
PRESSURE [PSI]          %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%%
ENTHALPY [MMBTU/LBMOL]  %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%%

------------------------------------------------------------------------------
SPLITTER TEAR STREAM REPORT CREATED BY SPEEDUP ON %%%%%%%%%%, %%%%%%%%%
Page 1 of 2

                        ************************
                        * C4 -> C2 TEAR STREAM *
                        ************************

STREAM NAME:            IN           OUT          SLACK
COMPONENT FLOWS [MOLE FRACTION]
  H2                    %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%%
  METHANE               %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%%
  ETHANE                %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%%
  PROPANE               %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%%
  IBUTANE               %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%%
  BUTANE                %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%%
  IPENTANE              %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%%
  PENTANE               %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%%
  BP45T65               %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%%
  BP65T85               %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%%
```

BP85T105	%%%%%%%%%%%%	%%%%%%%%%%%%	%%%%%%%%%%%%
BP105T125	%%%%%%%%%%%%	%%%%%%%%%%%%	%%%%%%%%%%%%
BP125T145	%%%%%%%%%%%%	%%%%%%%%%%%%	%%%%%%%%%%%%
BP145T165	%%%%%%%%%%%%	%%%%%%%%%%%%	%%%%%%%%%%%%
BP165T185	%%%%%%%%%%%%	%%%%%%%%%%%%	%%%%%%%%%%%%
BP185T205	%%%%%%%%%%%%	%%%%%%%%%%%%	%%%%%%%%%%%%
BP205T235	%%%%%%%%%%%%	%%%%%%%%%%%%	%%%%%%%%%%%%
BP235T265	%%%%%%%%%%%%	%%%%%%%%%%%%	%%%%%%%%%%%%
BP265T295	%%%%%%%%%%%%	%%%%%%%%%%%%	%%%%%%%%%%%%
BP295T325	%%%%%%%%%%%%	%%%%%%%%%%%%	%%%%%%%%%%%%
BP325T355	%%%%%%%%%%%%	%%%%%%%%%%%%	%%%%%%%%%%%%
BP355T395	%%%%%%%%%%%%	%%%%%%%%%%%%	%%%%%%%%%%%%
BP395T510	%%%%%%%%%%%%	%%%%%%%%%%%%	%%%%%%%%%%%%
MOLE FLOW [LBMOL/HR]	%%%%%%%%%%%%	%%%%%%%%%%%%	%%%%%%%%%%%%
TEMPERATURE [F]	%%%%%%%%%%%%	%%%%%%%%%%%%	%%%%%%%%%%%%
PRESSURE [PSI]	%%%%%%%%%%%%	%%%%%%%%%%%%	%%%%%%%%%%%%
ENTHALPY [MMBTU/LBMOL]	%%%%%%%%%%%%	%%%%%%%%%%%%	%%%%%%%%%%%%

SPLITTER TEAR STREAM REPORT CREATED BY SPEEDUP ON %%%%%%%%%%%, %%%%%%%%%%

****
* C5 -> C2 TEAR STREAM *
****

STREAM NAME:	IN	OUT	SLACK
COMPONENT FLOWS [MOLE FRACTION]			
H2	%%%%%%%%%%%%	%%%%%%%%%%%%	%%%%%%%%%%%%
METHANE	%%%%%%%%%%%%	%%%%%%%%%%%%	%%%%%%%%%%%%
ETHANE	%%%%%%%%%%%%	%%%%%%%%%%%%	%%%%%%%%%%%%
PROPANE	%%%%%%%%%%%%	%%%%%%%%%%%%	%%%%%%%%%%%%
IBUTANE	%%%%%%%%%%%%	%%%%%%%%%%%%	%%%%%%%%%%%%
BUTANE	%%%%%%%%%%%%	%%%%%%%%%%%%	%%%%%%%%%%%%
IPENTANE	%%%%%%%%%%%%	%%%%%%%%%%%%	%%%%%%%%%%%%
PENTANE	%%%%%%%%%%%%	%%%%%%%%%%%%	%%%%%%%%%%%%
BP45T65	%%%%%%%%%%%%	%%%%%%%%%%%%	%%%%%%%%%%%%
BP65T85	%%%%%%%%%%%%	%%%%%%%%%%%%	%%%%%%%%%%%%
BP85T105	%%%%%%%%%%%%	%%%%%%%%%%%%	%%%%%%%%%%%%
BP105T125	%%%%%%%%%%%%	%%%%%%%%%%%%	%%%%%%%%%%%%
BP125T145	%%%%%%%%%%%%	%%%%%%%%%%%%	%%%%%%%%%%%%
BP145T165	%%%%%%%%%%%%	%%%%%%%%%%%%	%%%%%%%%%%%%
BP165T185	%%%%%%%%%%%%	%%%%%%%%%%%%	%%%%%%%%%%%%
BP185T205	%%%%%%%%%%%%	%%%%%%%%%%%%	%%%%%%%%%%%%
BP205T235	%%%%%%%%%%%%	%%%%%%%%%%%%	%%%%%%%%%%%%
BP235T265	%%%%%%%%%%%%	%%%%%%%%%%%%	%%%%%%%%%%%%

```
  BP265T295             %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%%
  BP295T325             %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%%
  BP325T355             %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%%
  BP355T395             %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%%
  BP395T510             %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%%
MOLE FLOW [LBMOL/HR]    %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%%
TEMPERATURE [F]         %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%%
PRESSURE [PSI]          %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%%
ENTHALPY [MMBTU/LBMOL] %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%%

--------------------------------------------------------------------------------
SPLITTER TEAR STREAM REPORT CREATED BY SPEEDUP ON %%%%%%%%%%, %%%%%%%%%
Page 1 of 2

                    ********************************
                    * C2REBL -> C2BT24 TEAR STREAM *
                    ********************************

STREAM NAME:              IN            OUT           SLACK
COMPONENT FLOWS [MOLE FRACTION]
  H2                    %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%%
  METHANE               %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%%
  ETHANE                %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%%
  PROPANE               %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%%
  IBUTANE               %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%%
  BUTANE                %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%%
  IPENTANE              %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%%
  PENTANE               %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%%
  BP45T65               %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%%
  BP65T85               %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%%
  BP85T105              %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%%
  BP105T125             %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%%
  BP125T145             %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%%
  BP145T165             %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%%
  BP165T185             %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%%
  BP185T205             %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%%
  BP205T235             %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%%
  BP235T265             %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%%
  BP265T295             %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%%
  BP295T325             %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%%
  BP325T355             %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%%
  BP355T395             %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%%
  BP395T510             %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%%
MOLE FLOW [LBMOL/HR]    %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%%
TEMPERATURE [F]         %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%%
PRESSURE [PSI]          %%%%%%%%%%%% %%%%%%%%%%%% %%%%%%%%%%%%
```

```
ENTHALPY [MMBTU/LBMOL] %%%%%%%%%%%%% %%%%%%%%%%%%% %%%%%%%%%%%%%

------------------------------------------------------------------------------
****
REPORT SIMSET
HELP
  Simulation Settings to be passed onto next simulation
  (Based on DRSIMSET.DAT or OPSIMSET.DAT in Aspen runs).
$ENDHELP
FIELDS
# Date and Time for report
  &1 = DATE ;
  &2 = TIME ;
  DECIMALS = 4
  NOBLANKS
# SpeedUp Variable Values
  &3 = C1FT14.F_IN ;
  &4 = C1FT14.T_IN ;
  &5 = MFEED.P_OUT ;
  &6 = C1COND.Q ;
  &7 = C1SPL1.L_OUT1 ;
  &8 = C1BT29.L_OUT2 ;
  &9 = C1BT29.L_OUT1 ;
  &10 = C1BT29.P - "C1S.TRAY_SS(2)".P ;
  &11 = C1COND.P ;
  &12 = "C1S.TRAY_SS(2)".P ;
  &13 = C2FT21.T_IN ;
  &14 = E1.P_OUT ;
  &15 = C2COND.T_DROP ;
  &16 = C2COND.Q ;
  &17 = C2BT24.L_OUT2 ;
  &18 = C2BT24.L_OUT1 ;
  &19 = C2BT24.P - "C2S1.TRAY_SS(2)".P ;
  &20 = C2COND.P ;
  &21 = "C2S1.TRAY_SS(2)".P ;
  &22 = C2SPL1.L_OUT1 ;
  &23 = C2SPL2.L_OUT1 ;
  &24 = C4REBL.L_OUT ;
  &25 = C5REBL.L_OUT ;
# Aspen Variable Values
  &26 = MFEED.F_OUT ;
  &27 = MFEED.T_OUT ;
  &28 = MFEED.P_OUT ;
  &29 = F12.L_OUT / (F13.L_OUT + F11.V_OUT) ;
  &30 = F11.V_OUT / (F13.L_OUT + F11.V_OUT) ;
```

```
  &31 = C1BT29.L_OUT2 ;
  &32 = C1REBL.T_OUT ;
  &33 = C1BT29.P - "C1S.TRAY_SS(2)".P ;
  &34 = C1COND.P ;
  &35 = "C1S.TRAY_SS(2)".P ;
  &36 = E1.T_OUT ;
  &37 = E1.P_OUT ;
  &38 = C2COND.L_OUT2 ;
  &39 = C2BT24.L_OUT2 ;
  &40 = C2REBL.T_OUT ;
  &41 = C2COND.T ;
  &42 = C2BT24.P - "C2S1.TRAY_SS(2)".P ;
  &43 = C2COND.P ;
  &44 = "C2S1.TRAY_SS(2)".P ;
  &45 = C2SPL1.L_OUT1 ;
  &46 = C2SPL2.L_OUT1 ;
  &47 = C4REBL.Q ;
  &48 = C5REBL.Q ;

DISPLAY
SIMULATION SETTINGS SUMMARY REPORT CREATED BY SPEEDUP ON %%%%%%%%%%, %%%%%%%%
Page 1 of 1

                  ********************************
                  * SPEEDUP SIMULATION SETTINGS *
                  ********************************

            STAB MOLEFLOW  [LBMOL/HR]   %%%%%%%%%%%%%%%%%
                STAB TEMP  [F]          %%%%%%%%%%%%%%%%%
               STAB PRESS  [PSIA]       %%%%%%%%%%%%%%%%%
             C1 COND DUTY  [MMBTU/HR]   %%%%%%%%%%%%%%%%%
             DIST-C4 FLOW  [LBMOL/HR]   %%%%%%%%%%%%%%%%%
             C1 REBL FLOW  [LBMOL/HR]   %%%%%%%%%%%%%%%%%
            SPLTFEED FLOW  [LBMOL/HR]   %%%%%%%%%%%%%%%%%
            C1 COL P DROP  [PSI]        %%%%%%%%%%%%%%%%%
            C1 TOP TRAY P  [PSIA]       %%%%%%%%%%%%%%%%%
            C1 2ND TRAY P  [PSIA]       %%%%%%%%%%%%%%%%%
              E1 OUT TEMP  [F]          %%%%%%%%%%%%%%%%%
             E1 OUT PRESS  [PSIA]       %%%%%%%%%%%%%%%%%
    C2 COND SC TEMP DROP   [F]          %%%%%%%%%%%%%%%%%
             C2 COND DUTY  [MMBTU/HR]   %%%%%%%%%%%%%%%%%
             C2 REBL FLOW  [LBMOL/HR]   %%%%%%%%%%%%%%%%%
              SSFEED FLOW  [LBMOL/HR]   %%%%%%%%%%%%%%%%%
            C2 COL P DROP  [PSI]        %%%%%%%%%%%%%%%%%
            C2 TOP TRAY P  [PSIA]       %%%%%%%%%%%%%%%%%
```

```
C2 2ND TRAY P  [PSIA]        %%%%%%%%%%%%%%%%
C2-C4 IC FLOW  [LBMOL/HR]    %%%%%%%%%%%%%%%%
C2-C5 IC FLOW  [LBMOL/HR]    %%%%%%%%%%%%%%%%
  HV-ISO FLOW  [LBMOL/HR]    %%%%%%%%%%%%%%%%
 LT-DIST FLOW  [LBMOL/HR]    %%%%%%%%%%%%%%%%

        ******************************
        * ASPEN SIMULATION SETTINGS *
        ******************************

STAB MOLEFLOW  [LBMOL/HR]    %%%%%%%%%%%%%%%%
    STAB TEMP  [F]           %%%%%%%%%%%%%%%%
   STAB PRESS  [PSIA]        %%%%%%%%%%%%%%%%
C1 RFLX RATIO  [MOL/MOL]     %%%%%%%%%%%%%%%%
  C1 VAP/DIST  [MOL/MOL]     %%%%%%%%%%%%%%%%
 C1 REBL FLOW  [LBMOL/HR]    %%%%%%%%%%%%%%%%
 C1 REBL TEMP  [F]           %%%%%%%%%%%%%%%%
C1 COL P DROP  [PSI]         %%%%%%%%%%%%%%%%
C1 TOP TRAY P  [PSIA]        %%%%%%%%%%%%%%%%
C1 2ND TRAY P  [PSIA]        %%%%%%%%%%%%%%%%
  E1 OUT TEMP  [F]           %%%%%%%%%%%%%%%%
 E1 OUT PRESS  [PSIA]        %%%%%%%%%%%%%%%%
 C2 DIST FLOW  [LBMOL/HR]    %%%%%%%%%%%%%%%%
 C2 REBL FLOW  [LBMOL/HR]    %%%%%%%%%%%%%%%%
 C2 REBL TEMP  [F]           %%%%%%%%%%%%%%%%
 C2 COND SC T  [F]           %%%%%%%%%%%%%%%%
C2 COL P DROP  [PSI]         %%%%%%%%%%%%%%%%
C2 TOP TRAY P  [PSIA]        %%%%%%%%%%%%%%%%
C2 2ND TRAY P  [PSIA]        %%%%%%%%%%%%%%%%
C2-C4 IC FLOW  [LBMOL/HR]    %%%%%%%%%%%%%%%%
C2-C5 IC FLOW  [LBMOL/HR]    %%%%%%%%%%%%%%%%
 C4 REBL DUTY  [MMBTU/HR]    %%%%%%%%%%%%%%%%
 C5 REBL DUTY  [MMBTU/HR]    %%%%%%%%%%%%%%%%
```

```
****
REPORT DREC
HELP
  Objective Function and Variable Summaries for the Data
  Reconciliation Runs.
$ENDHELP
# Text Prcessor Setup
  ?SET nocomp=24 ?END
FIELDS
# Date and Time for report
```

```
  &1 = DATE ;
  &2 = TIME ;
  DECIMALS = 4
  NOBLANKS
# Variable Values
  &3 = F01.STDVOLFL * GLOBAL.CNVF2B ;
  &6 = F11.STDVOLFV ;
  &9 = F12.STDVOLFL * GLOBAL.CNVF2B ;
  &12 = F13.STDVOLFL * GLOBAL.CNVF2B ;
  &15 = F21.STDVOLFL * GLOBAL.CNVF2B ;
  &18 = F22.STDVOLFL * GLOBAL.CNVF2B ;
  &21 = F31.STDVOLFL * GLOBAL.CNVF2B ;
  &24 = F32.STDVOLFL * GLOBAL.CNVF2B ;
  &27 = F41.STDVOLFL * GLOBAL.CNVF2B ;
  &30 = F42.STDVOLFL * GLOBAL.CNVF2B ;
  &33 = F51.STDVOLFL * GLOBAL.CNVF2B ;
  &36 = F52.STDVOLFL * GLOBAL.CNVF2B ;
  &39 = F61.STDVOLFL * GLOBAL.CNVF2B ;
  &42 = F62.STDVOLFL * GLOBAL.CNVF2B ;
  &45 = MFEED.T_OUT ;
  &48 = "C1S.TRAY_SS(2)".T ;
  &51 = C1COND.T ;
  &54 = C1BT29.TL_OUT1 ;
  &57 = C1REBL.T_OUT ;
  &60 = E1.T_OUT ;
  &63 = "C2S1.TRAY_SS(2)".T ;
  &66 = C2COND.T ;
  &69 = C2SPL1.TL_OUT1 ;
  &72 = C4REBL.TL_OUT ;
  &75 = C2SPL2.TL_OUT1 ;
  &78 = C5REBL.TL_OUT ;
  &81 = C2BT24.TL_OUT1 ;
  &84 = C2REBL.T_OUT ;
# Offset Values
  &4 = (F01.STDVOLFL * GLOBAL.CNVF2B) - GLOBAL.SF01 ;
  &7 = F11.STDVOLFV - GLOBAL.SF11 ;
  &10 = (F12.STDVOLFL * GLOBAL.CNVF2B) - GLOBAL.SF12 ;
  &13 = (F13.STDVOLFL * GLOBAL.CNVF2B) - GLOBAL.SF13 ;
  &16 = (F21.STDVOLFL * GLOBAL.CNVF2B) - GLOBAL.SF21 ;
  &19 = (F22.STDVOLFL * GLOBAL.CNVF2B) - GLOBAL.SF22 ;
  &22 = (F31.STDVOLFL * GLOBAL.CNVF2B) - GLOBAL.SF31 ;
  &25 = (F32.STDVOLFL * GLOBAL.CNVF2B) - GLOBAL.SF32 ;
  &28 = (F41.STDVOLFL * GLOBAL.CNVF2B) - GLOBAL.SF41 ;
  &31 = (F42.STDVOLFL * GLOBAL.CNVF2B) - GLOBAL.SF42 ;
  &34 = (F51.STDVOLFL * GLOBAL.CNVF2B) - GLOBAL.SF51 ;
```

```
  &37 = (F52.STDVOLFL * GLOBAL.CNVF2B) - GLOBAL.SF52 ;
  &40 = (F61.STDVOLFL * GLOBAL.CNVF2B) - GLOBAL.SF61 ;
  &43 = (F62.STDVOLFL * GLOBAL.CNVF2B) - GLOBAL.SF62 ;
  &46 = MFEED.T_OUT - GLOBAL.ST01 ;
  &49 = "C1S.TRAY_SS(2)".T - GLOBAL.ST12 ;
  &52 = C1COND.T - GLOBAL.ST13 ;
  &55 = C1BT29.TL_OUT1 - GLOBAL.ST21 ;
  &58 = C1REBL.T_OUT - GLOBAL.ST22 ;
  &61 = E1.T_OUT - GLOBAL.ST23 ;
  &64 = "C2S1.TRAY_SS(2)".T - GLOBAL.ST31 ;
  &67 = C2COND.T - GLOBAL.ST32 ;
  &70 = C2SPL1.TL_OUT1 - GLOBAL.ST41 ;
  &73 = C4REBL.TL_OUT - GLOBAL.ST42 ;
  &76 = C2SPL2.TL_OUT1 - GLOBAL.ST51 ;
  &79 = C5REBL.TL_OUT - GLOBAL.ST52 ;
  &82 = C2BT24.TL_OUT1 - GLOBAL.ST61 ;
  &85 = C2REBL.T_OUT - GLOBAL.ST62 ;
# Error Values
  &5 = GLOBAL.EF01 ;
  &8 = GLOBAL.EF11 ;
  &11 = GLOBAL.EF12 ;
  &14 = GLOBAL.EF13 ;
  &17 = GLOBAL.EF21 ;
  &20 = GLOBAL.EF22 ;
  &23 = GLOBAL.EF31 ;
  &26 = GLOBAL.EF32 ;
  &29 = GLOBAL.EF41 ;
  &32 = GLOBAL.EF42 ;
  &35 = GLOBAL.EF51 ;
  &38 = GLOBAL.EF52 ;
  &41 = GLOBAL.EF61 ;
  &44 = GLOBAL.EF62 ;
  &47 = GLOBAL.ET01 ;
  &50 = GLOBAL.ET12 ;
  &53 = GLOBAL.ET13 ;
  &56 = GLOBAL.ET21 ;
  &59 = GLOBAL.ET22 ;
  &62 = GLOBAL.ET23 ;
  &65 = GLOBAL.ET31 ;
  &68 = GLOBAL.ET32 ;
  &71 = GLOBAL.ET41 ;
  &74 = GLOBAL.ET42 ;
  &77 = GLOBAL.ET51 ;
  &80 = GLOBAL.ET52 ;
  &83 = GLOBAL.ET61 ;
```

```
  &86 = GLOBAL.ET62 ;
# Objective Function Summary
  &87 = &5 + &8 + &11 + &14 + &17 + &20 + &23 + &26 + &29 + &32 + &35
      + &38 + &41 + &44 + &47 + &50 + &53 + &56 + &59 + &62 + &65 + &68
      + &71 + &74 + &77 + &80 + &83 + &86 ;
DISPLAY
DATA RECONCILIATION SUMMARY REPORT CREATED BY SPEEDUP ON %%%%%%%%%%, %%%%%%%%%
Page 1 of 1
                    ************************************
                    * VARIABLE, OFFSET & ERROR SUMMARY *
                    ************************************

                        VARIABLE        OFFSET          ERROR

       F01 [BBL/DAY]    %%%%%%%%%%%%%   %%%%%%%%%%%%%   %%%%%%%%%%%%%
       F11 [CUFT/HR]    %%%%%%%%%%%%%   %%%%%%%%%%%%%   %%%%%%%%%%%%%
       F12 [BBL/DAY]    %%%%%%%%%%%%%   %%%%%%%%%%%%%   %%%%%%%%%%%%%
       F13 [BBL/DAY]    %%%%%%%%%%%%%   %%%%%%%%%%%%%   %%%%%%%%%%%%%
       F21 [BBL/DAY]    %%%%%%%%%%%%%   %%%%%%%%%%%%%   %%%%%%%%%%%%%
       F22 [BBL/DAY]    %%%%%%%%%%%%%   %%%%%%%%%%%%%   %%%%%%%%%%%%%
       F31 [BBL/DAY]    %%%%%%%%%%%%%   %%%%%%%%%%%%%   %%%%%%%%%%%%%
       F32 [BBL/DAY]    %%%%%%%%%%%%%   %%%%%%%%%%%%%   %%%%%%%%%%%%%
       F41 [BBL/DAY]    %%%%%%%%%%%%%   %%%%%%%%%%%%%   %%%%%%%%%%%%%
       F42 [BBL/DAY]    %%%%%%%%%%%%%   %%%%%%%%%%%%%   %%%%%%%%%%%%%
       F51 [BBL/DAY]    %%%%%%%%%%%%%   %%%%%%%%%%%%%   %%%%%%%%%%%%%
       F52 [BBL/DAY]    %%%%%%%%%%%%%   %%%%%%%%%%%%%   %%%%%%%%%%%%%
       F61 [BBL/DAY]    %%%%%%%%%%%%%   %%%%%%%%%%%%%   %%%%%%%%%%%%%
       F62 [BBL/DAY]    %%%%%%%%%%%%%   %%%%%%%%%%%%%   %%%%%%%%%%%%%
       T01 [F]          %%%%%%%%%%%%%   %%%%%%%%%%%%%   %%%%%%%%%%%%%
       T12 [F]          %%%%%%%%%%%%%   %%%%%%%%%%%%%   %%%%%%%%%%%%%
       T13 [F]          %%%%%%%%%%%%%   %%%%%%%%%%%%%   %%%%%%%%%%%%%
       T21 [F]          %%%%%%%%%%%%%   %%%%%%%%%%%%%   %%%%%%%%%%%%%
       T22 [F]          %%%%%%%%%%%%%   %%%%%%%%%%%%%   %%%%%%%%%%%%%
       T23 [F]          %%%%%%%%%%%%%   %%%%%%%%%%%%%   %%%%%%%%%%%%%
       T31 [F]          %%%%%%%%%%%%%   %%%%%%%%%%%%%   %%%%%%%%%%%%%
       T32 [F]          %%%%%%%%%%%%%   %%%%%%%%%%%%%   %%%%%%%%%%%%%
       T41 [F]          %%%%%%%%%%%%%   %%%%%%%%%%%%%   %%%%%%%%%%%%%
       T42 [F]          %%%%%%%%%%%%%   %%%%%%%%%%%%%   %%%%%%%%%%%%%
       T51 [F]          %%%%%%%%%%%%%   %%%%%%%%%%%%%   %%%%%%%%%%%%%
       T52 [F]          %%%%%%%%%%%%%   %%%%%%%%%%%%%   %%%%%%%%%%%%%
       T61 [F]          %%%%%%%%%%%%%   %%%%%%%%%%%%%   %%%%%%%%%%%%%
       T62 [F]          %%%%%%%%%%%%%   %%%%%%%%%%%%%   %%%%%%%%%%%%%

                    ******************************
                    * OBJECTIVE FUNCTION SUMMARY *
```

```
                    ******************************

FINAL OBJECTIVE FUNCTION VALUE = %%%%%%%%%%%%

--------------------------------------------------------------------------------
****
REPORT OPT
HELP
  Objective Function and Constraint Summaries for the Optimization
  Simulation Run.
$ENDHELP
# Text Prcessor Setup
  ?SET nocomp=24 ?END
FIELDS
# Date and Time for report
  &1 = DATE ;
  &2 = TIME ;
  &31 = DATE ;
  &32 = TIME ;
  &61 = DATE ;
  &62 = TIME ;
  DECIMALS = 6
  NOBLANKS
# Flow Constraint Values
  &3 = F12.STDVOLFL * GLOBAL.CNVF2B ;
  &4 = F32.STDVOLFL * GLOBAL.CNVF2B ;
  &5 = F41.STDVOLFL * GLOBAL.CNVF2B ;
  &6 = F51.STDVOLFL * GLOBAL.CNVF2B ;
# Temperature Constraint Values
  &7 = "C1S.TRAY_SS(3)".T ;
  &8 = C1REBL.T_OUT ;
  &9 = "C2S1.TRAY_SS(2)".T ;
  &10 = C2REBL.T_OUT ;
# C5 Composition Constraint Values
  &11 = F13.C5S * 100 ;
# RVP Constraint Values
  &12 = RVP31.RVP ;
  &13 = C2COND.T_BUBBLE ;
# TBP Constraint Values
  &14 = F42.TBP10 ;
  &15 = F42.TBP90 ;
  &16 = F52.TBP10 ;
  &17 = F52.TBP90 ;
# Objective Function Summary
  &18 = GLOBAL.PVALS - GLOBAL.CSTAB - GLOBAL.COF1 - GLOBAL.COF2 ;
```

```
  &19 = GLOBAL.PVALS ;
  &20 = GLOBAL.CSTAB ;
  &21 = GLOBAL.COF1 + GLOBAL.COF2 ;
# Stream Std-vol Flows
  &22 = F01.STDVOLFL ;
  &24 = F11.STDVOLFL ;
  &26 = F13.STDVOLFL ;
  &27 = F31.STDVOLFL ;
  &28 = F42.STDVOLFL ;
  &29 = F52.STDVOLFL ;
  &30 = F62.STDVOLFL ;
# Reboiler Heat Duties
  &23 = C1REBL.Q ;
  &25 = C2REBL.Q ;
# Variable Values
  &33 = F01.STDVOLFL * GLOBAL.CNVF2B ;
  &34 = F11.STDVOLFV ;
  &35 = F12.STDVOLFL * GLOBAL.CNVF2B ;
  &36 = F13.STDVOLFL * GLOBAL.CNVF2B ;
  &37 = F21.STDVOLFL * GLOBAL.CNVF2B ;
  &38 = F22.STDVOLFL * GLOBAL.CNVF2B ;
  &39 = F31.STDVOLFL * GLOBAL.CNVF2B ;
  &40 = F32.STDVOLFL * GLOBAL.CNVF2B ;
  &41 = F41.STDVOLFL * GLOBAL.CNVF2B ;
  &42 = F42.STDVOLFL * GLOBAL.CNVF2B ;
  &43 = F51.STDVOLFL * GLOBAL.CNVF2B ;
  &44 = F52.STDVOLFL * GLOBAL.CNVF2B ;
  &45 = F61.STDVOLFL * GLOBAL.CNVF2B ;
  &46 = F62.STDVOLFL * GLOBAL.CNVF2B ;
  &47 = MFEED.T_OUT ;
  &48 = "C1S.TRAY_SS(2)".T ;
  &49 = C1COND.T ;
  &50 = C1BT29.TL_OUT1 ;
  &51 = C1REBL.T_OUT ;
  &52 = E1.T_OUT ;
  &53 = "C2S1.TRAY_SS(2)".T ;
  &54 = C2COND.T ;
  &55 = C2SPL1.TL_OUT1 ;
  &56 = C4REBL.TL_OUT ;
  &57 = C2SPL2.TL_OUT1 ;
  &58 = C5REBL.TL_OUT ;
  &59 = C2BT24.TL_OUT1 ;
  &60 = C2REBL.T_OUT ;
# Set Point Values
  &63 = F12.STDVOLFL * GLOBAL.CNVF2B ;
```

```
  &64 = "C1S.TRAY_SS(3)".T ;
  &65 = F13.C5S ;
  &66 = C1REBL.T_OUT ;
  &67 = F32.STDVOLFL * GLOBAL.CNVF2B ;
  &68 = F41.STDVOLFL * GLOBAL.CNVF2B ;
  &69 = F51.STDVOLFL * GLOBAL.CNVF2B ;
  &70 = C2REBL.T_OUT ;
  &71 = F42.TBP10 ;
  &72 = F42.TBP90 ;
  &73 = F52.TBP10 ;
  &74 = F52.TBP90 ;

DISPLAY
OPTIMIZATION SUMMARY REPORT CREATED BY SPEEDUP ON %%%%%%%%%%%, %%%%%%%%%%
Page 1 of 3
                        ****************************
                        * CONSTRAINT VALUE SUMMARY *
                        ****************************

VARIABLE                LOWER BOUND  VALUE              UPPER BOUND

F12 [BBL/DAY]                5500.0  %%%%%%%%%%%%%%%%        9000.0
F32 [BBL/DAY]                        %%%%%%%%%%%%%%%%       25000.0
F41 [BBL/DAY]                        %%%%%%%%%%%%%%%%       10604.0
F51 [BBL/DAY]                        %%%%%%%%%%%%%%%%        6000.0

T11 [F]                       125.0  %%%%%%%%%%%%%%%%         220.0
T22 [F]                       500.0  %%%%%%%%%%%%%%%%         675.0
T31 [F]                       120.0  %%%%%%%%%%%%%%%%         250.0
T62 [F]                       550.0  %%%%%%%%%%%%%%%%         700.0

A11 [PERCENT C5's]              0.5  %%%%%%%%%%%%%%%%          15.0

A31-RVP [PSI]                        %%%%%%%%%%%%%%%%          14.0
A31-TBUBBLE [F]               110.0  %%%%%%%%%%%%%%%%

A41-TBP @ .10 VOL [F]         176.0  %%%%%%%%%%%%%%%%         266.0
A41-TBP @ .90 VOL [F]         284.0  %%%%%%%%%%%%%%%%         464.0

A51-TBP @ .10 VOL [F]         302.0  %%%%%%%%%%%%%%%%         554.0
A51-TBP @ .90 VOL [F]         302.0  %%%%%%%%%%%%%%%%         554.0


                       ******************************
                       * OBJECTIVE FUNCTION SUMMARY *
```

****

FINAL OBJECTIVE FUNCTION VALUE (PROFIT) [$/HR] = %%%%%%%%%%%%%%%%

TOTAL PRODUCT VALUES [$/HR] = %%%%%%%%%%%%%%%%
TOTAL FEED COSTS [$/HR] = %%%%%%%%%%%%%%%%
TOTAL UTILITY COSTS [$/HR] = %%%%%%%%%%%%%%%%

STREAM	STD-VOL FLOWS [BBL/DAY]	COLUMN	HEAT DUTY [MMBTU/HR]
STAB	%%%%%%%%%%%%%%%%%%	C1	%%%%%%%%%%%%%%%%%%
VAP	%%%%%%%%%%%%%%%%%%	C2	%%%%%%%%%%%%%%%%%%
DIST-C4	%%%%%%%%%%%%%%%%%%		
LT-ISO	%%%%%%%%%%%%%%%%%%		
HV-ISO	%%%%%%%%%%%%%%%%%%		
LT-DIST	%%%%%%%%%%%%%%%%%%		
SSFEED	%%%%%%%%%%%%%%%%%%		

****
* VARIABLE SUMMARY *
****

	VARIABLE
F01 [BBL/DAY]	%%%%%%%%%%%%
F11 [CUFT/HR]	%%%%%%%%%%%%
F12 [BBL/DAY]	%%%%%%%%%%%%
F13 [BBL/DAY]	%%%%%%%%%%%%
F21 [BBL/DAY]	%%%%%%%%%%%%
F22 [BBL/DAY]	%%%%%%%%%%%%
F31 [BBL/DAY]	%%%%%%%%%%%%
F32 [BBL/DAY]	%%%%%%%%%%%%
F41 [BBL/DAY]	%%%%%%%%%%%%
F42 [BBL/DAY]	%%%%%%%%%%%%
F51 [BBL/DAY]	%%%%%%%%%%%%
F52 [BBL/DAY]	%%%%%%%%%%%%
F61 [BBL/DAY]	%%%%%%%%%%%%
F62 [BBL/DAY]	%%%%%%%%%%%%
T01 [F]	%%%%%%%%%%%%
T12 [F]	%%%%%%%%%%%%
T13 [F]	%%%%%%%%%%%%

```
      T21 [F]          %%%%%%%%%%%%
      T22 [F]          %%%%%%%%%%%%
      T23 [F]          %%%%%%%%%%%%
      T31 [F]          %%%%%%%%%%%%
      T32 [F]          %%%%%%%%%%%%
      T41 [F]          %%%%%%%%%%%%
      T42 [F]          %%%%%%%%%%%%
      T51 [F]          %%%%%%%%%%%%
      T52 [F]          %%%%%%%%%%%%
      T61 [F]          %%%%%%%%%%%%
      T62 [F]          %%%%%%%%%%%%

------------------------------------------------------------------------------
OPTIMIZATION SUMMARY REPORT CREATED BY SPEEDUP ON %%%%%%%%%%%, %%%%%%%%%%
Page 3 of 3
                          **********************
                          * SET POINT SUMMARY *
                          **********************

      C1 REFLUX FLOW (F12) [BBL/DAY]                  = %%%%%%%%%%%%%%%%
      C1 TRAY 3 TEMPERATURE (T11) [F]                 = %%%%%%%%%%%%%%%%
      DIST-C4 C5'S MOLE FRACTION (A11) [LBMOL/LBMOL] = %%%%%%%%%%%%%%%%
      C1 REBOILER TEMPERATURE (T22) [F]               = %%%%%%%%%%%%%%%%
      C2 REFLUX FLOW (F32) [BBL/DAY]                  = %%%%%%%%%%%%%%%%
      C4 FEED FLOW (F41) [BBL/DAY]                    = %%%%%%%%%%%%%%%%
      C5 FEED FLOW (F51) [BBL/DAY]                    = %%%%%%%%%%%%%%%%
      C2 REBOILER TEMPERATURE (T62) [F]               = %%%%%%%%%%%%%%%%
      HV-ISO .10 TBP TEMPERATURE (A41) [F]            = %%%%%%%%%%%%%%%%
      HV-ISO .90 TBP TEMPERATURE (A41) [F]            = %%%%%%%%%%%%%%%%
      LT-DIST .10 TBP TEMPERATURE (A51) [F]           = %%%%%%%%%%%%%%%%
      LT-DIST .90 TBP TEMPERATURE (A51) [F]           = %%%%%%%%%%%%%%%%
```

E.10 prop6.inp

The following ASPEN PLUS input code, `prop6.inp`, contains PROPERTIES PLUS properties information for use with both `rtodr6.speedup` and `rtopt6.speedup`.

```
;
;Input Summary created by ASPEN PLUS Rel. 9.2-1 at 11:10:58 Tue Jul 30, 1996
;Directory C:\MATT  Filename C:\MATT\prop6.inp
;
```

```
PROP

TITLE 'STABILIZER-SPLITTER SIMULATION PROPERTIES (PROP6)'

IN-UNITS ENG

DIAGNOSTICS
    HISTORY SYS-LEVEL=1 SIM-LEVEL=1 PROP-LEVEL=0 STREAM-LEVEL=0  &
        CONV-LEVEL=0 COST-LEVEL=0 ECON-LEVEL=0

SIM-OPTIONS FREE-WATER=NO

DESCRIPTION "
        PROPERTIES FOR SPEEDUP RUNS STBSPL6*, DREC6*, AND OPT6* "

DATABANKS ASPENPCD  / PURE856  / PURECOMP  / AQUEOUS  /  &
        SOLIDS  / INORGANIC

PROP-SOURCES ASPENPCD  / PURE856  / PURECOMP  / AQUEOUS  /  &
        SOLIDS  / INORGANIC

COMPONENTS
    H2 H2 H2 /
    METHANE CH4 METHANE /
    ETHANE C2H6 ETHANE /
    PROPANE C3H8 PROPANE /
    IBUTANE C4H10-2 IBUTANE /
    BUTANE C4H10-1 BUTANE /
    IPENTANE C5H12-2 IPENTANE /
    PENTANE C5H12-1 PENTANE /
    B45T65 * B45T65 /
    B65T85 * B65T85 /
    B85T105 * B85T105 /
    B105T125 * B105T125 /
    B125T145 * B125T145 /
    B145T165 * B145T165 /
    B165T185 * B165T185 /
    B185T205 * B185T205 /
    B205T235 * B205T235 /
    B235T265 * B235T265 /
    B265T295 * B265T295 /
    B295T325 * B295T325 /
    B325T355 * B325T355 /
    B355T395 * B355T395 /
```

```
    B395T510 * B395T510 /
    AIR AIR AIR

PC-USER
    PC-DEF API-METH B45T65 NBP=128.49380 GRAV=.64840  &
        MW=85.98380
    PC-DEF API-METH B65T85 NBP=167.34120 GRAV=.69060  &
        MW=89.15430
    PC-DEF API-METH B85T105 NBP=203.78650 GRAV=.71370  &
        MW=99.91430
    PC-DEF API-METH B105T125 NBP=239.16230 GRAV=.7320  &
        MW=110.25780
    PC-DEF API-METH B125T145 NBP=275.02460 GRAV=.74760  &
        MW=120.44870
    PC-DEF API-METH B145T165 NBP=310.94660 GRAV=.76280  &
        MW=132.08530
    PC-DEF API-METH B165T185 NBP=347.01010 GRAV=.77630  &
        MW=144.87180
    PC-DEF API-METH B185T205 NBP=383.08250 GRAV=.78560  &
        MW=160.93490
    PC-DEF API-METH B205T235 NBP=428.20310 GRAV=.79940  &
        MW=180.13470
    PC-DEF API-METH B235T265 NBP=482.06570 GRAV=.81410  &
        MW=204.11590
    PC-DEF API-METH B265T295 NBP=537.76350 GRAV=.8240  &
        MW=236.50410
    PC-DEF API-METH B295T325 NBP=590.42290 GRAV=.82590  &
        MW=270.94860
    PC-DEF API-METH B325T355 NBP=643.56240 GRAV=.83250  &
        MW=307.94140
    PC-DEF API-METH B355T395 NBP=704.2320 GRAV=.83780  &
        MW=354.40350
    PC-DEF API-METH B395T510 NBP=786.51950 GRAV=.85560  &
        MW=418.92730

PROPERTIES PENG-ROB SYSOPO SOLU-WATER=2

REPORT NOCOSTBLOCK NOUNITS NOUTILITIES NOECONOMIC

PROPERTY-REP NOCOMPS NOPARAMS NOPCES
```

E.11 rtodr6setup.cmd

The following SPEEDUP commands, `rtodr6setup.cmd`, are passed to SPEEDUP when the program `setup` is setting up the rtodr6 data reconciliation simulation.

```
2
new rtodr
store rtodr6
store model6x
store report6y
report
noexecute
opt
run
quit
```

E.12 rtodr6run.cmd

The following SPEEDUP commands, `rtodr6run.cmd`, are passed to SPEEDUP when the program `run` is running the rtodr6 data reconciliation simulation.

```
store drpreset
opt
use
run
res drpreset
preset c*
preset e*
preset f*
preset m*
preset r*
end
save oppreset (OW)
delete result drpreset
dump oppreset (=result)
disp
file
rep drec
rep streamsum
```

```
rep c1
rep c2
global
term
end
quit
```

E.13 rtopt6setup.cmd

The following SPEEDUP commands, `rtopt6setup.cmd`, are passed to SPEEDUP when the program `setup` is setting up the rtopt6 optimization simulation.

```
2
new rtopt
store rtopt6
store model6x
store report6y
report
noexecute
opt
run
quit
```

E.14 rtopt6run.cmd

The following SPEEDUP commands, `rtopt6run.cmd`, are passed to SPEEDUP when the program `run` is setting up the rtopt6 optimization simulation.

```
store oppreset
opt
use
run
res oppreset
preset c*
preset e*
preset f*
preset m*
preset r*
end
save drpreset (OW)
```

```
delete result oppreset
dump drpreset (=result)
disp
file
rep opt
rep streamsum
rep c1
rep c2
global
term
end
quit
```

NOTE TO USERS

Page(s) missing in number only; text follows. Microfilmed as received.

464

UMI

Appendix F

Plant Simulation Code

F.1 plant6.inp

The following ASPEN PLUS input code, textttplant6.inp, performs a steady state plant simulation.

```
;
;Input Summary created by ASPEN PLUS Rel. 9.2-1 at 13:21:06 Wed May 14, 1997
;Directory D:\  Filename plant6.inp
;

TITLE 'STABILIZER-SPLITTER PLANT SIMULATION'

IN-UNITS ENG VOLUME-FLOW='BBL/DAY' ENTHALPY-FLO='MMBTU/HR'  &
        VOLUME=BBL HEAD=FT HEAT=MMBTU

DEF-STREAMS CONVEN ALL

DIAGNOSTICS
    HISTORY SYS-LEVEL=4 SIM-LEVEL=2 PROP-LEVEL=0 STREAM-LEVEL=1  &
        CONV-LEVEL=2 COST-LEVEL=0 ECON-LEVEL=0

SIM-OPTIONS
    IN-UNITS ENG
    SIM-OPTIONS FREE-WATER=NO

RUN-CONTROL MAX-TIME=1.0000E+08 MAX-ERRORS=9999 MAX-FORT-ERR=50
```

```
DESCRIPTION "
        STABILIZER (C1): MODELLED BY RADFRAC WITH 29 STAGES WITH A PUMPAROUND
          REBOILER.
        SPLITTER (C2, C4, C5): MODELLED BY PETROFRAC WITH A 24 STAGE MAIN
          COLUMN WITH AN EXTERNAL REBOILER, AND TWO 3 STAGE REBOILED
          SIDESTRIPPERS (C4 AND C5). EXTERNAL REBOILER CONSISTS OF AN FSPLIT
          AND A HEATER.
        PROPERTIES: 15 PSEUDO-COMPONENTS, 8 CONVENTIONAL COMPONENTS,
          PENG-ROBINSON E.O.S. (N.B. WATER IS NOT USED - NO FREE WATER CALCS).
        BASIS: MOLE BASIS USED (LBMOL/HR) "

DATABANKS ASPENPCD  / PURE856  / PURECOMP  / AQUEOUS  /  &
        SOLIDS  / INORGANIC

PROP-SOURCES ASPENPCD  / PURE856  / PURECOMP  / AQUEOUS  /  &
        SOLIDS  / INORGANIC

COMPONENTS
    H2 H2 H2 /
    METHANE CH4 METHANE /
    ETHANE C2H6 ETHANE /
    PROPANE C3H8 PROPANE /
    IBUTANE C4H10-2 IBUTANE /
    BUTANE C4H10-1 BUTANE /
    IPENTANE C5H12-2 IPENTANE /
    PENTANE C5H12-1 PENTANE /
    B45T65 * B45T65 /
    B65T85 * B65T85 /
    B85T105 * B85T105 /
    B105T125 * B105T125 /
    B125T145 * B125T145 /
    B145T165 * B145T165 /
    B165T185 * B165T185 /
    B185T205 * B185T205 /
    B205T235 * B205T235 /
    B235T265 * B235T265 /
    B265T295 * B265T295 /
    B295T325 * B295T325 /
    B325T355 * B325T355 /
    B355T395 * B355T395 /
    B395T510 * B395T510

PC-USER
    PC-DEF API-METH B45T65 NBP=128.49380 GRAV=.64840  &
        MW=85.98380
```

```
    PC-DEF API-METH B65T85 NBP=167.34120 GRAV=.69060  &
        MW=89.15430
    PC-DEF API-METH B85T105 NBP=203.78650 GRAV=.71370  &
        MW=99.91430
    PC-DEF API-METH B105T125 NBP=239.16230 GRAV=.7320  &
        MW=110.25780
    PC-DEF API-METH B125T145 NBP=275.02460 GRAV=.74760  &
        MW=120.44870
    PC-DEF API-METH B145T165 NBP=310.94660 GRAV=.76280  &
        MW=132.08530
    PC-DEF API-METH B165T185 NBP=347.01010 GRAV=.77630  &
        MW=144.87180
    PC-DEF API-METH B185T205 NBP=383.08250 GRAV=.78560  &
        MW=160.93490
    PC-DEF API-METH B205T235 NBP=428.20310 GRAV=.79940  &
        MW=180.13470
    PC-DEF API-METH B235T265 NBP=482.06570 GRAV=.81410  &
        MW=204.11590
    PC-DEF API-METH B265T295 NBP=537.76350 GRAV=.8240  &
        MW=236.50410
    PC-DEF API-METH B295T325 NBP=590.42290 GRAV=.82590  &
        MW=270.94860
    PC-DEF API-METH B325T355 NBP=643.56240 GRAV=.83250  &
        MW=307.94140
    PC-DEF API-METH B355T395 NBP=704.2320 GRAV=.83780  &
        MW=354.40350
    PC-DEF API-METH B395T510 NBP=786.51950 GRAV=.85560  &
        MW=418.92730

FLOWSHEET
    BLOCK C1 IN=STAB OUT=VAP DIST-C4 SPLTFEED C1RFLX C1REBL
    BLOCK C2C4C5 IN=SPLT C2REBL OUT=LT-ISO C2BOT HV-ISO  &
        LT-DIST C2RFLX IC1 IC2
    BLOCK E1 IN=SPLTFEED OUT=SPLT
    BLOCK F2 IN=C2RBFD OUT=C2REBL
    BLOCK C2SPL1 IN=C2BOT OUT=C2RBFD SSFEED

PROPERTIES PENG-ROB SOLU-WATER=2

PROP-SET RVP REIDVP SUBSTREAM=MIXED

PROP-SET STDVOL VLSTDMX UNITS='BBL/DAY' SUBSTREAM=MIXED PHASE=T

PROP-SET TBP-10 TBPT SUBSTREAM=MIXED LVPCT=10.0
```

```
PROP-SET TBP-90 TBPT SUBSTREAM=MIXED LVPCT=90.0

PROP-SET TBUB TBUB SUBSTREAM=MIXED

PROP-SET V-STDVOL VMX UNITS='CUFT/HR' SUBSTREAM=MIXED PHASE=V  &
        TEMP=60.0 PRES=14.6960

PROP-SET VOL VMX UNITS='CUFT/HR' SUBSTREAM=MIXED PHASE=T

STREAM C2REBL
    SUBSTREAM MIXED TEMP=683.899910 PRES=35.65810  &
        MOLE-FLOW=1327.2170
    MOLE-FRAC H2 0.0 / METHANE 1.3091E-18 / ETHANE  &
        3.9177E-15 / PROPANE 3.4897E-11 / IBUTANE 2.3334E-08 /  &
        BUTANE 5.2249E-08 / IPENTANE 1.0680E-06 / PENTANE  &
        6.7323E-07 / B45T65 5.3548E-06 / B65T85 .000013843 /  &
        B85T105 .000042311 / B105T125 .000123760 / B125T145  &
        .000363910 / B145T165 .00117030 / B165T185 .00455110 /  &
        B185T205 .0187060 / B205T235 .1040 / B235T265 .14430  &
        / B265T295 .16840 / B295T325 .17710 / B325T355  &
        .15840 / B355T395 .14950 / B395T510 .073060

STREAM STAB
    SUBSTREAM MIXED TEMP=412.954698 PRES=200.0  &
        MOLE-FLOW=1487.17710
    MOLE-FRAC H2 .000703680 / METHANE .0246830 / ETHANE  &
        .0473640 / PROPANE .0387240 / IBUTANE .0535310 /  &
        BUTANE .0312790 / IPENTANE .0351930 / PENTANE  &
        .0161490 / B45T65 .0472560 / B65T85 .0435620 /  &
        B85T105 .0496490 / B105T125 .0530710 / B125T145  &
        .0523260 / B145T165 .0489630 / B165T185 .0444180 /  &
        B185T205 .041240 / B205T235 .0607690 / B235T265  &
        .0537820 / B265T295 .0598310 / B295T325 .0626870 /  &
        B325T355 .056050 / B355T395 .0529210 / B395T510  &
        .0258480

BLOCK C2SPL1 FSPLIT
    DESCRIPTION "C2 BOTTOMS FLOW SPLITTER"
    MOLE-FLOW C2RBFD 1327.2170

BLOCK E1 HEATER
    IN-UNITS ENG VOLUME-FLOW='BBL/DAY' ENTHALPY-FLO='MMBTU/HR'
    DESCRIPTION "E1 HEAT EXCHANGER"
    PARAM TEMP=479.65650 PRES=34.0
```

```
BLOCK F2 HEATER
    DESCRIPTION "C2 REBOILER FURNACE"
    PARAM TEMP=683.899910 PRES=0.0

BLOCK C1 RADFRAC
    SUBOBJECTS PUMPAROUND = C1REBL
    DESCRIPTION "STABILISER"
    PARAM NSTAGE=29 NPA=1
    PUMPAROUND C1REBL 29 29 NPHASE=2 MOLE-FLOW=2005.69940  &
        TEMP=586.084880
    FEEDS STAB 14 ON-STAGE
    PRODUCTS VAP 1 V / DIST-C4 1 L / SPLTFEED 29 L
    PSEUDO-STREA C1RFLX 1 / C1REBL PA=C1REBL
    P-SPEC 1 163.0 / 2 167.0
    COL-SPECS QN=0.0 DP-COL=6.0 MOLE-RDV=.60262460  &
        MOLE-RR=3.15630830
    T-EST 1 89.5350 / 2 149.040 / 3 169.260 / 4 182.040 /  &
        5 192.140 / 6 200.260 / 7 206.640 / 8 211.760 /  &
        9 216.380 / 10 221.630 / 11 229.350 / 12 243.370 /  &
        13 275.160 / 14 386.670 / 15 389.440 / 16 391.110  &
        / 17 392.160 / 18 392.890 / 19 393.490 / 20  &
        394.060 / 21 394.670 / 22 395.410 / 23 396.360 /  &
        24 397.680 / 25 399.710 / 26 403.380 / 27 411.850  &
        / 28 436.50 / 29 519.830
    L-EST 1 1022.0250 / 2 991.26280 / 3 989.080 / 4  &
        974.47260 / 5 962.86240 / 6 955.53050 / 7 949.85040  &
        / 8 942.2590 / 9 928.30430 / 10 900.91910 / 11  &
        845.60520 / 12 723.95590 / 13 413.96660 / 14  &
        1449.9490 / 15 1500.8610 / 16 1531.4830 / 17  &
        1550.3670 / 18 1563.3520 / 19 1573.6410 / 20  &
        1583.0950 / 21 1593.0030 / 22 1604.5030 / 23  &
        1618.8610 / 24 1637.8240 / 25 1664.3390 / 26  &
        1703.9790 / 27 1764.4650 / 28 1818.9030 / 29  &
        3205.2810
    V-EST 1 173.31230 / 2 1195.3370 / 3 1278.8590 / 4  &
        1276.6760 / 5 1262.0680 / 6 1250.4580 / 7 1243.1260  &
        / 8 1237.4460 / 9 1229.8550 / 10 1215.90 / 11  &
        1188.5150 / 12 1133.2010 / 13 1011.5520 / 14  &
        701.56250 / 15 250.3680 / 16 301.27930 / 17  &
        331.90150 / 18 350.78570 / 19 363.77040 / 20  &
        374.06010 / 21 383.51360 / 22 393.42180 / 23  &
        404.92140 / 24 419.27950 / 25 438.24310 / 26  &
        464.75790 / 27 504.39750 / 28 564.88340 / 29  &
        619.32170
    TRAY-REPORT TRAY-OPTION=ALL-TRAYS
```

```
    REPORT NOCOMPS NOHYDRAULIC NOENTH NOSPLITS

BLOCK C2C4C5 PETROFRAC
    SUBOBJECTS STRIPPER = C4 C5
    DESCRIPTION "SPLITTER"
    PARAM NPA=0 NSTRIP=2 NSSTRIP=6 NSTAGE=24 HYDRAULIC=NO
    FEEDS SPLT 21 ON-STAGE / C2REBL 24 ON-STAGE
    PRODUCTS LT-ISO 1 L / C2BOT 24 L
    COL-SPECS CONDENSER=SUBCOOLED REBOILER=NONE-BOTFEED  &
        T1=84.7225280 MOLE-D=186.57490 DP-COL=12.70
    P-SPEC 1 18.95810 / 2 22.95810
    T-EST 1 84.7230 / 2 150.930 / 3 169.420 / 4 181.130 /  &
        5 188.750 / 6 194.010 / 7 197.990 / 8 201.440 /  &
        9 204.930 / 10 208.970 / 11 214.20 / 12 221.490 /  &
        13 232.330 / 14 249.640 / 15 278.020 / 16 320.730  &
        / 17 358.980 / 18 388.210 / 19 415.850 / 20  &
        441.860 / 21 483.390 / 22 531.190 / 23 570.180 /  &
        24 626.450
    V-EST 1 0.0 / 2 2776.1290 / 3 3094.8350 / 4 3095.9990  &
        / 5 3101.710 / 6 3106.5220 / 7 3108.8040 / 8  &
        3107.4250 / 9 3100.7670 / 10 3086.2130 / 11  &
        3059.6610 / 12 3014.5630 / 13 2939.0650 / 14  &
        2809.8090 / 15 2473.5240 / 16 2206.0180 / 17  &
        2037.2320 / 18 1833.3290 / 19 1747.6360 / 20  &
        1647.220 / 21 1439.3880 / 22 772.39920 / 23  &
        965.84870 / 24 953.74740
    L-EST 1 2776.1290 / 2 2908.260 / 3 2909.4240 / 4  &
        2915.1350 / 5 2919.9470 / 6 2922.2290 / 7 2920.850  &
        / 8 2914.1920 / 9 2899.6380 / 10 2873.0860 / 11  &
        2827.9890 / 12 2752.490 / 13 2623.2340 / 14  &
        2410.6880 / 15 2143.1810 / 16 1496.7840 / 17  &
        1408.0990 / 18 1322.4060 / 19 973.77810 / 20  &
        765.94590 / 21 1298.5380 / 22 1491.9880 / 23  &
        1479.8870 / 24 1853.3340
    PSEUDO-STREA C2RFLX STAGE=1
    STRIPPER C4 NSTAGE=3 LDRAW=15 VRETURN=14 PRODUCT=HV-ISO  &
        MOLE-DRAW=477.61110 Q-REB=1.90090
    STRIPPER C5 NSTAGE=3 LDRAW=18 VRETURN=17 PRODUCT=LT-DIST  &
        MOLE-DRAW=248.21130 Q-REB=2.39920
    STR-T-EST C4 1 278.790 / C4 2 282.20 / C4 3 292.030 /  &
        C5 1 402.780 / C5 2 414.880 / C5 3 429.50
    STR-V-EST C4 1 123.70 / C4 2 122.70 / C4 3 123.40 /  &
        C5 1 115.20 / C5 2 125.60 / C5 3 130.40
    STR-L-EST C4 1 476.60 / C4 2 477.20 / C4 3 353.90 /  &
        C5 1 258.60 / C5 2 263.40 / C5 3 133.0
```

```
    STR-PSEUDO-S C4 IC1 SOURCE=FEED DRAW=15 / C5 IC2  &
        SOURCE=FEED DRAW=18
    TRAY-REPORT TRAY-OPTION=ALL-TRAYS
    REPORT RESULTS NOCOMPS NOHYDRAULIC NOENTH

FORTRAN DATA
F     LOGICAL THERE
    DEFINE F01 STREAM-VAR STREAM=STAB SUBSTREAM=MIXED  &
        VARIABLE=STDVOL-FLOW
    DEFINE F11 STREAM-PROP STREAM=VAP PROPERTY=V-STDVOL
    DEFINE F12 STREAM-VAR STREAM=C1RFLX SUBSTREAM=MIXED  &
        VARIABLE=STDVOL-FLOW
    DEFINE F13 STREAM-VAR STREAM=DIST-C4 SUBSTREAM=MIXED  &
        VARIABLE=STDVOL-FLOW
    DEFINE F21 STREAM-VAR STREAM=C1REBL SUBSTREAM=MIXED  &
        VARIABLE=STDVOL-FLOW
    DEFINE F22 STREAM-VAR STREAM=SPLTFEED SUBSTREAM=MIXED  &
        VARIABLE=STDVOL-FLOW
    DEFINE F31 STREAM-VAR STREAM=LT-ISO SUBSTREAM=MIXED  &
        VARIABLE=STDVOL-FLOW
    DEFINE F32 STREAM-VAR STREAM=C2RFLX SUBSTREAM=MIXED  &
        VARIABLE=STDVOL-FLOW
    DEFINE F41 STREAM-VAR STREAM=IC1 SUBSTREAM=MIXED  &
        VARIABLE=STDVOL-FLOW
    DEFINE F42 STREAM-VAR STREAM=HV-ISO SUBSTREAM=MIXED  &
        VARIABLE=STDVOL-FLOW
    DEFINE F51 STREAM-VAR STREAM=IC2 SUBSTREAM=MIXED  &
        VARIABLE=STDVOL-FLOW
    DEFINE F52 STREAM-VAR STREAM=LT-DIST SUBSTREAM=MIXED  &
        VARIABLE=STDVOL-FLOW
    DEFINE F61 STREAM-VAR STREAM=C2REBL SUBSTREAM=MIXED  &
        VARIABLE=STDVOL-FLOW
    DEFINE F62 STREAM-VAR STREAM=SSFEED SUBSTREAM=MIXED  &
        VARIABLE=STDVOL-FLOW
    DEFINE T01 STREAM-VAR STREAM=STAB SUBSTREAM=MIXED  &
        VARIABLE=TEMP
    DEFINE T12 BLOCK-VAR BLOCK=C1 VARIABLE=TEMP  &
        SENTENCE=PROFILE ID1=2
    DEFINE T13 STREAM-VAR STREAM=DIST-C4 SUBSTREAM=MIXED  &
        VARIABLE=TEMP
    DEFINE T21 STREAM-VAR STREAM=SPLTFEED SUBSTREAM=MIXED  &
        VARIABLE=TEMP
    DEFINE T22 STREAM-VAR STREAM=C1REBL SUBSTREAM=MIXED  &
        VARIABLE=TEMP
    DEFINE T23 STREAM-VAR STREAM=SPLT SUBSTREAM=MIXED  &
```

```
        VARIABLE=TEMP
    DEFINE T31 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=TEMP  &
        SENTENCE=PROFILE ID1=2
    DEFINE T32 STREAM-VAR STREAM=LT-ISO SUBSTREAM=MIXED  &
        VARIABLE=TEMP
    DEFINE T41 STREAM-VAR STREAM=IC1 SUBSTREAM=MIXED  &
        VARIABLE=TEMP
    DEFINE T42 STREAM-VAR STREAM=HV-ISO SUBSTREAM=MIXED  &
        VARIABLE=TEMP
    DEFINE T51 STREAM-VAR STREAM=IC2 SUBSTREAM=MIXED  &
        VARIABLE=TEMP
    DEFINE T52 STREAM-VAR STREAM=LT-DIST SUBSTREAM=MIXED  &
        VARIABLE=TEMP
    DEFINE T61 STREAM-VAR STREAM=SSFEED SUBSTREAM=MIXED  &
        VARIABLE=TEMP
    DEFINE T62 STREAM-VAR STREAM=C2REBL SUBSTREAM=MIXED  &
        VARIABLE=TEMP
C WRITE TO FILE MEASURED VARIABLES VALUES AFTER DATA REC
F     WRITE(NTERM,*) '+++ PLANT RESULTS FORTRAN STARTED +++'
F     INQUIRE (FILE='PLVARS.DAT', EXIST=THERE)
F     IF (THERE) THEN
F       WRITE(NTERM,*) '*** ERROR: PLVARS.DAT ALREADY EXISITS ***'
F       WRITE(NRPT,*) '*** ERROR: PLVARS.DAT ALREADY EXISTED ***'
F       WRITE(NHSTRY,*) '*** ERROR: PLVARS.DAT ALREADY EXISTS ***'
F       GO TO 90
F     END IF
F     OPEN (UNIT=63, STATUS='NEW', FILE='PLVARS.DAT')
F     WRITE(63,100) 'F01', F01
F     WRITE(63,100) 'F11', F11
F     WRITE(63,100) 'F12', F12
F     WRITE(63,100) 'F13', F13
F     WRITE(63,100) 'F21', F21
F     WRITE(63,100) 'F22', F22
F     WRITE(63,100) 'F31', F31
F     WRITE(63,100) 'F32', F32
F     WRITE(63,100) 'F41', F41
F     WRITE(63,100) 'F42', F42
F     WRITE(63,100) 'F51', F51
F     WRITE(63,100) 'F52', F52
F     WRITE(63,100) 'F61', F61
F     WRITE(63,100) 'F62', F62
F     WRITE(63,100) 'T01', T01
F     WRITE(63,100) 'T12', T12
F     WRITE(63,100) 'T13', T13
F     WRITE(63,100) 'T21', T21
```

```
F     WRITE(63,100) 'T22', T22
F     WRITE(63,100) 'T23', T23
F     WRITE(63,100) 'T31', T31
F     WRITE(63,100) 'T32', T32
F     WRITE(63,100) 'T41', T41
F     WRITE(63,100) 'T42', T42
F     WRITE(63,100) 'T51', T51
F     WRITE(63,100) 'T52', T52
F     WRITE(63,100) 'T61', T61
F     WRITE(63,100) 'T62', T62
F     CLOSE (UNIT=63)
F  90 WRITE(NTERM,*) '+++ PLANT RESULTS FORTRAN COMPLETED +++'
C FORMAT STATEMENTS
F 100 FORMAT(A13, 2X, E15.8)
    EXECUTE LAST

FORTRAN SPCSET
F     LOGICAL THERE
    DEFINE STABF STREAM-VAR STREAM=STAB SUBSTREAM=MIXED  &
        VARIABLE=MOLE-FLOW
    DEFINE STABT STREAM-VAR STREAM=STAB SUBSTREAM=MIXED  &
        VARIABLE=TEMP
    DEFINE STABP STREAM-VAR STREAM=STAB SUBSTREAM=MIXED  &
        VARIABLE=PRES
    DEFINE STBZ1 MOLE-FRAC STREAM=STAB SUBSTREAM=MIXED  &
        COMPONENT=H2
    DEFINE STBZ2 MOLE-FRAC STREAM=STAB SUBSTREAM=MIXED  &
        COMPONENT=METHANE
    DEFINE STBZ3 MOLE-FRAC STREAM=STAB SUBSTREAM=MIXED  &
        COMPONENT=ETHANE
    DEFINE STBZ4 MOLE-FRAC STREAM=STAB SUBSTREAM=MIXED  &
        COMPONENT=PROPANE
    DEFINE STBZ5 MOLE-FRAC STREAM=STAB SUBSTREAM=MIXED  &
        COMPONENT=IBUTANE
    DEFINE STBZ6 MOLE-FRAC STREAM=STAB SUBSTREAM=MIXED  &
        COMPONENT=BUTANE
    DEFINE STBZ7 MOLE-FRAC STREAM=STAB SUBSTREAM=MIXED  &
        COMPONENT=IPENTANE
    DEFINE STBZ8 MOLE-FRAC STREAM=STAB SUBSTREAM=MIXED  &
        COMPONENT=PENTANE
    DEFINE STBZ9 MOLE-FRAC STREAM=STAB SUBSTREAM=MIXED  &
        COMPONENT=B45T65
    DEFINE STBZ10 MOLE-FRAC STREAM=STAB SUBSTREAM=MIXED  &
        COMPONENT=B65T85
    DEFINE STBZ11 MOLE-FRAC STREAM=STAB SUBSTREAM=MIXED  &
```

```
    COMPONENT=B85T105
DEFINE STBZ12 MOLE-FRAC STREAM=STAB SUBSTREAM=MIXED  &
    COMPONENT=B105T125
DEFINE STBZ13 MOLE-FRAC STREAM=STAB SUBSTREAM=MIXED  &
    COMPONENT=B125T145
DEFINE STBZ14 MOLE-FRAC STREAM=STAB SUBSTREAM=MIXED  &
    COMPONENT=B145T165
DEFINE STBZ15 MOLE-FRAC STREAM=STAB SUBSTREAM=MIXED  &
    COMPONENT=B165T185
DEFINE STBZ16 MOLE-FRAC STREAM=STAB SUBSTREAM=MIXED  &
    COMPONENT=B185T205
DEFINE STBZ17 MOLE-FRAC STREAM=STAB SUBSTREAM=MIXED  &
    COMPONENT=B205T235
DEFINE STBZ18 MOLE-FRAC STREAM=STAB SUBSTREAM=MIXED  &
    COMPONENT=B235T265
DEFINE STBZ19 MOLE-FRAC STREAM=STAB SUBSTREAM=MIXED  &
    COMPONENT=B265T295
DEFINE STBZ20 MOLE-FRAC STREAM=STAB SUBSTREAM=MIXED  &
    COMPONENT=B295T325
DEFINE STBZ21 MOLE-FRAC STREAM=STAB SUBSTREAM=MIXED  &
    COMPONENT=B325T355
DEFINE STBZ22 MOLE-FRAC STREAM=STAB SUBSTREAM=MIXED  &
    COMPONENT=B355T395
DEFINE STBZ23 MOLE-FRAC STREAM=STAB SUBSTREAM=MIXED  &
    COMPONENT=B395T510
DEFINE C2RBF1 BLOCK-VAR BLOCK=C2SPL1 SENTENCE=MOLE-FLOW  &
    VARIABLE=FLOW ID1=C2RBFD
DEFINE C2RBF2 STREAM-VAR STREAM=C2REBL SUBSTREAM=MIXED  &
    VARIABLE=MOLE-FLOW
DEFINE C2RBT1 BLOCK-VAR BLOCK=F2 VARIABLE=TEMP  &
    SENTENCE=PARAM
DEFINE C2RBT2 STREAM-VAR STREAM=C2REBL SUBSTREAM=MIXED  &
    VARIABLE=TEMP
DEFINE C2RBP1 BLOCK-VAR BLOCK=F2 VARIABLE=PRES  &
    SENTENCE=PARAM
DEFINE C2RBP2 STREAM-VAR STREAM=C2REBL SUBSTREAM=MIXED  &
    VARIABLE=PRES
DEFINE C2RZ1 MOLE-FRAC STREAM=C2REBL SUBSTREAM=MIXED  &
    COMPONENT=H2
DEFINE C2RZ2 MOLE-FRAC STREAM=C2REBL SUBSTREAM=MIXED  &
    COMPONENT=METHANE
DEFINE C2RZ3 MOLE-FRAC STREAM=C2REBL SUBSTREAM=MIXED  &
    COMPONENT=ETHANE
DEFINE C2RZ4 MOLE-FRAC STREAM=C2REBL SUBSTREAM=MIXED  &
    COMPONENT=PROPANE
```

```
DEFINE C2RZ5 MOLE-FRAC STREAM=C2REBL SUBSTREAM=MIXED  &
    COMPONENT=IBUTANE
DEFINE C2RZ6 MOLE-FRAC STREAM=C2REBL SUBSTREAM=MIXED  &
    COMPONENT=BUTANE
DEFINE C2RZ7 MOLE-FRAC STREAM=C2REBL SUBSTREAM=MIXED  &
    COMPONENT=IPENTANE
DEFINE C2RZ8 MOLE-FRAC STREAM=C2REBL SUBSTREAM=MIXED  &
    COMPONENT=PENTANE
DEFINE C2RZ9 MOLE-FRAC STREAM=C2REBL SUBSTREAM=MIXED  &
    COMPONENT=B45T65
DEFINE C2RZ10 MOLE-FRAC STREAM=C2REBL SUBSTREAM=MIXED  &
    COMPONENT=B65T85
DEFINE C2RZ11 MOLE-FRAC STREAM=C2REBL SUBSTREAM=MIXED  &
    COMPONENT=B85T105
DEFINE C2RZ12 MOLE-FRAC STREAM=C2REBL SUBSTREAM=MIXED  &
    COMPONENT=B105T125
DEFINE C2RZ13 MOLE-FRAC STREAM=C2REBL SUBSTREAM=MIXED  &
    COMPONENT=B125T145
DEFINE C2RZ14 MOLE-FRAC STREAM=C2REBL SUBSTREAM=MIXED  &
    COMPONENT=B145T165
DEFINE C2RZ15 MOLE-FRAC STREAM=C2REBL SUBSTREAM=MIXED  &
    COMPONENT=B165T185
DEFINE C2RZ16 MOLE-FRAC STREAM=C2REBL SUBSTREAM=MIXED  &
    COMPONENT=B185T205
DEFINE C2RZ17 MOLE-FRAC STREAM=C2REBL SUBSTREAM=MIXED  &
    COMPONENT=B205T235
DEFINE C2RZ18 MOLE-FRAC STREAM=C2REBL SUBSTREAM=MIXED  &
    COMPONENT=B235T265
DEFINE C2RZ19 MOLE-FRAC STREAM=C2REBL SUBSTREAM=MIXED  &
    COMPONENT=B265T295
DEFINE C2RZ20 MOLE-FRAC STREAM=C2REBL SUBSTREAM=MIXED  &
    COMPONENT=B295T325
DEFINE C2RZ21 MOLE-FRAC STREAM=C2REBL SUBSTREAM=MIXED  &
    COMPONENT=B325T355
DEFINE C2RZ22 MOLE-FRAC STREAM=C2REBL SUBSTREAM=MIXED  &
    COMPONENT=B355T395
DEFINE C2RZ23 MOLE-FRAC STREAM=C2REBL SUBSTREAM=MIXED  &
    COMPONENT=B395T510
DEFINE E1T BLOCK-VAR BLOCK=E1 VARIABLE=TEMP SENTENCE=PARAM
DEFINE E1P BLOCK-VAR BLOCK=E1 VARIABLE=PRES SENTENCE=PARAM
DEFINE C1RR BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-RR  &
    SENTENCE=COL-SPECS
DEFINE C1RDV BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-RDV  &
    SENTENCE=COL-SPECS
DEFINE C1RBFL BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
```

```
    SENTENCE=PUMPAROUND ID1=C1REBL
DEFINE C1RBT BLOCK-VAR BLOCK=C1 VARIABLE=TEMP  &
    SENTENCE=PUMPAROUND ID1=C1REBL
DEFINE C1P1 BLOCK-VAR BLOCK=C1 VARIABLE=PRES  &
    SENTENCE=P-SPEC ID1=1
DEFINE C1P2 BLOCK-VAR BLOCK=C1 VARIABLE=PRES  &
    SENTENCE=P-SPEC ID1=2
DEFINE C1DP BLOCK-VAR BLOCK=C1 VARIABLE=DP-COL  &
    SENTENCE=COL-SPECS
DEFINE C2MD BLOCK-VAR BLOCK=C2C4C5 VARIABLE=MOLE-D  &
    SENTENCE=COL-SPECS
DEFINE C2SC BLOCK-VAR BLOCK=C2C4C5 VARIABLE=T1  &
    SENTENCE=COL-SPECS
DEFINE C2P1 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=PRES  &
    SENTENCE=P-SPEC ID1=1
DEFINE C2P2 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=PRES  &
    SENTENCE=P-SPEC ID1=2
DEFINE C2DP BLOCK-VAR BLOCK=C2C4C5 VARIABLE=DP-COL  &
    SENTENCE=COL-SPECS
DEFINE C2IC1 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=MOLE-DRAW  &
    SENTENCE=STRIPPER ID1=C4
DEFINE C2IC2 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=MOLE-DRAW  &
    SENTENCE=STRIPPER ID1=C5
DEFINE C4QN BLOCK-VAR BLOCK=C2C4C5 VARIABLE=Q-REB  &
    SENTENCE=STRIPPER ID1=C4
DEFINE C5QN BLOCK-VAR BLOCK=C2C4C5 VARIABLE=Q-REB  &
    SENTENCE=STRIPPER ID1=C5
DEFINE C1T1 BLOCK-VAR BLOCK=C1 VARIABLE=TEMP SENTENCE=T-EST  &
    ID1=1
DEFINE C1T2 BLOCK-VAR BLOCK=C1 VARIABLE=TEMP SENTENCE=T-EST  &
    ID1=2
DEFINE C1T3 BLOCK-VAR BLOCK=C1 VARIABLE=TEMP SENTENCE=T-EST  &
    ID1=3
DEFINE C1T4 BLOCK-VAR BLOCK=C1 VARIABLE=TEMP SENTENCE=T-EST  &
    ID1=4
DEFINE C1T5 BLOCK-VAR BLOCK=C1 VARIABLE=TEMP SENTENCE=T-EST  &
    ID1=5
DEFINE C1T6 BLOCK-VAR BLOCK=C1 VARIABLE=TEMP SENTENCE=T-EST  &
    ID1=6
DEFINE C1T7 BLOCK-VAR BLOCK=C1 VARIABLE=TEMP SENTENCE=T-EST  &
    ID1=7
DEFINE C1T8 BLOCK-VAR BLOCK=C1 VARIABLE=TEMP SENTENCE=T-EST  &
    ID1=8
DEFINE C1T9 BLOCK-VAR BLOCK=C1 VARIABLE=TEMP SENTENCE=T-EST  &
    ID1=9
```

```
DEFINE C1T10 BLOCK-VAR BLOCK=C1 VARIABLE=TEMP  &
    SENTENCE=T-EST ID1=10
DEFINE C1T11 BLOCK-VAR BLOCK=C1 VARIABLE=TEMP  &
    SENTENCE=T-EST ID1=11
DEFINE C1T12 BLOCK-VAR BLOCK=C1 VARIABLE=TEMP  &
    SENTENCE=T-EST ID1=12
DEFINE C1T13 BLOCK-VAR BLOCK=C1 VARIABLE=TEMP  &
    SENTENCE=T-EST ID1=13
DEFINE C1T14 BLOCK-VAR BLOCK=C1 VARIABLE=TEMP  &
    SENTENCE=T-EST ID1=14
DEFINE C1T15 BLOCK-VAR BLOCK=C1 VARIABLE=TEMP  &
    SENTENCE=T-EST ID1=15
DEFINE C1T16 BLOCK-VAR BLOCK=C1 VARIABLE=TEMP  &
    SENTENCE=T-EST ID1=16
DEFINE C1T17 BLOCK-VAR BLOCK=C1 VARIABLE=TEMP  &
    SENTENCE=T-EST ID1=17
DEFINE C1T18 BLOCK-VAR BLOCK=C1 VARIABLE=TEMP  &
    SENTENCE=T-EST ID1=18
DEFINE C1T19 BLOCK-VAR BLOCK=C1 VARIABLE=TEMP  &
    SENTENCE=T-EST ID1=19
DEFINE C1T20 BLOCK-VAR BLOCK=C1 VARIABLE=TEMP  &
    SENTENCE=T-EST ID1=20
DEFINE C1T21 BLOCK-VAR BLOCK=C1 VARIABLE=TEMP  &
    SENTENCE=T-EST ID1=21
DEFINE C1T22 BLOCK-VAR BLOCK=C1 VARIABLE=TEMP  &
    SENTENCE=T-EST ID1=22
DEFINE C1T23 BLOCK-VAR BLOCK=C1 VARIABLE=TEMP  &
    SENTENCE=T-EST ID1=23
DEFINE C1T24 BLOCK-VAR BLOCK=C1 VARIABLE=TEMP  &
    SENTENCE=T-EST ID1=24
DEFINE C1T25 BLOCK-VAR BLOCK=C1 VARIABLE=TEMP  &
    SENTENCE=T-EST ID1=25
DEFINE C1T26 BLOCK-VAR BLOCK=C1 VARIABLE=TEMP  &
    SENTENCE=T-EST ID1=26
DEFINE C1T27 BLOCK-VAR BLOCK=C1 VARIABLE=TEMP  &
    SENTENCE=T-EST ID1=27
DEFINE C1T28 BLOCK-VAR BLOCK=C1 VARIABLE=TEMP  &
    SENTENCE=T-EST ID1=28
DEFINE C1T29 BLOCK-VAR BLOCK=C1 VARIABLE=TEMP  &
    SENTENCE=T-EST ID1=29
DEFINE C2T1 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=TEMP  &
    SENTENCE=T-EST ID1=1
DEFINE C2T2 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=TEMP  &
    SENTENCE=T-EST ID1=2
DEFINE C2T3 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=TEMP  &
```

```
    SENTENCE=T-EST ID1=3
DEFINE C2T4 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=TEMP  &
    SENTENCE=T-EST ID1=4
DEFINE C2T5 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=TEMP  &
    SENTENCE=T-EST ID1=5
DEFINE C2T6 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=TEMP  &
    SENTENCE=T-EST ID1=6
DEFINE C2T7 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=TEMP  &
    SENTENCE=T-EST ID1=7
DEFINE C2T8 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=TEMP  &
    SENTENCE=T-EST ID1=8
DEFINE C2T9 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=TEMP  &
    SENTENCE=T-EST ID1=9
DEFINE C2T10 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=TEMP  &
    SENTENCE=T-EST ID1=10
DEFINE C2T11 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=TEMP  &
    SENTENCE=T-EST ID1=11
DEFINE C2T12 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=TEMP  &
    SENTENCE=T-EST ID1=12
DEFINE C2T13 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=TEMP  &
    SENTENCE=T-EST ID1=13
DEFINE C2T14 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=TEMP  &
    SENTENCE=T-EST ID1=14
DEFINE C2T15 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=TEMP  &
    SENTENCE=T-EST ID1=15
DEFINE C2T16 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=TEMP  &
    SENTENCE=T-EST ID1=16
DEFINE C2T17 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=TEMP  &
    SENTENCE=T-EST ID1=17
DEFINE C2T18 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=TEMP  &
    SENTENCE=T-EST ID1=18
DEFINE C2T19 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=TEMP  &
    SENTENCE=T-EST ID1=19
DEFINE C2T20 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=TEMP  &
    SENTENCE=T-EST ID1=20
DEFINE C2T21 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=TEMP  &
    SENTENCE=T-EST ID1=21
DEFINE C2T22 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=TEMP  &
    SENTENCE=T-EST ID1=22
DEFINE C2T23 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=TEMP  &
    SENTENCE=T-EST ID1=23
DEFINE C2T24 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=TEMP  &
    SENTENCE=T-EST ID1=24
DEFINE C4T1 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=TEMP  &
    SENTENCE=STR-T-EST ID1=C4 ID2=1
```

```
DEFINE C4T2 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=TEMP  &
    SENTENCE=STR-T-EST ID1=C4 ID2=2
DEFINE C4T3 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=TEMP  &
    SENTENCE=STR-T-EST ID1=C4 ID2=3
DEFINE C5T1 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=TEMP  &
    SENTENCE=STR-T-EST ID1=C5 ID2=1
DEFINE C5T2 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=TEMP  &
    SENTENCE=STR-T-EST ID1=C5 ID2=2
DEFINE C5T3 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=TEMP  &
    SENTENCE=STR-T-EST ID1=C5 ID2=3
DEFINE C1L1 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=L-EST ID1=1
DEFINE C1L2 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=L-EST ID1=2
DEFINE C1L3 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=L-EST ID1=3
DEFINE C1L4 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=L-EST ID1=4
DEFINE C1L5 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=L-EST ID1=5
DEFINE C1L6 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=L-EST ID1=6
DEFINE C1L7 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=L-EST ID1=7
DEFINE C1L8 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=L-EST ID1=8
DEFINE C1L9 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=L-EST ID1=9
DEFINE C1L10 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=L-EST ID1=10
DEFINE C1L11 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=L-EST ID1=11
DEFINE C1L12 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=L-EST ID1=12
DEFINE C1L13 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=L-EST ID1=13
DEFINE C1L14 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=L-EST ID1=14
DEFINE C1L15 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=L-EST ID1=15
DEFINE C1L16 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=L-EST ID1=16
DEFINE C1L17 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=L-EST ID1=17
DEFINE C1L18 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
```

```
    SENTENCE=L-EST ID1=18
DEFINE C1L19 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=L-EST ID1=19
DEFINE C1L20 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=L-EST ID1=20
DEFINE C1L21 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=L-EST ID1=21
DEFINE C1L22 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=L-EST ID1=22
DEFINE C1L23 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=L-EST ID1=23
DEFINE C1L24 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=L-EST ID1=24
DEFINE C1L25 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=L-EST ID1=25
DEFINE C1L26 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=L-EST ID1=26
DEFINE C1L27 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=L-EST ID1=27
DEFINE C1L28 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=L-EST ID1=28
DEFINE C1L29 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=L-EST ID1=29
DEFINE C1V1 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=V-EST ID1=1
DEFINE C1V2 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=V-EST ID1=2
DEFINE C1V3 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=V-EST ID1=3
DEFINE C1V4 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=V-EST ID1=4
DEFINE C1V5 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=V-EST ID1=5
DEFINE C1V6 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=V-EST ID1=6
DEFINE C1V7 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=V-EST ID1=7
DEFINE C1V8 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=V-EST ID1=8
DEFINE C1V9 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=V-EST ID1=9
DEFINE C1V10 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=V-EST ID1=10
DEFINE C1V11 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=V-EST ID1=11
```

```
DEFINE C1V12 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=V-EST ID1=12
DEFINE C1V13 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=V-EST ID1=13
DEFINE C1V14 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=V-EST ID1=14
DEFINE C1V15 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=V-EST ID1=15
DEFINE C1V16 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=V-EST ID1=16
DEFINE C1V17 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=V-EST ID1=17
DEFINE C1V18 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=V-EST ID1=18
DEFINE C1V19 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=V-EST ID1=19
DEFINE C1V20 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=V-EST ID1=20
DEFINE C1V21 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=V-EST ID1=21
DEFINE C1V22 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=V-EST ID1=22
DEFINE C1V23 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=V-EST ID1=23
DEFINE C1V24 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=V-EST ID1=24
DEFINE C1V25 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=V-EST ID1=25
DEFINE C1V26 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=V-EST ID1=26
DEFINE C1V27 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=V-EST ID1=27
DEFINE C1V28 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=V-EST ID1=28
DEFINE C1V29 BLOCK-VAR BLOCK=C1 VARIABLE=MOLE-FLOW  &
    SENTENCE=V-EST ID1=29
DEFINE C2L1 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
    SENTENCE=L-EST ID1=1
DEFINE C2L2 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
    SENTENCE=L-EST ID1=2
DEFINE C2L3 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
    SENTENCE=L-EST ID1=3
DEFINE C2L4 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
    SENTENCE=L-EST ID1=4
DEFINE C2L5 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
```

```
    SENTENCE=L-EST ID1=5
DEFINE C2L6 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
    SENTENCE=L-EST ID1=6
DEFINE C2L7 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
    SENTENCE=L-EST ID1=7
DEFINE C2L8 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
    SENTENCE=L-EST ID1=8
DEFINE C2L9 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
    SENTENCE=L-EST ID1=9
DEFINE C2L10 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
    SENTENCE=L-EST ID1=10
DEFINE C2L11 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
    SENTENCE=L-EST ID1=11
DEFINE C2L12 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
    SENTENCE=L-EST ID1=12
DEFINE C2L13 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
    SENTENCE=L-EST ID1=13
DEFINE C2L14 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
    SENTENCE=L-EST ID1=14
DEFINE C2L15 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
    SENTENCE=L-EST ID1=15
DEFINE C2L16 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
    SENTENCE=L-EST ID1=16
DEFINE C2L17 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
    SENTENCE=L-EST ID1=17
DEFINE C2L18 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
    SENTENCE=L-EST ID1=18
DEFINE C2L19 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
    SENTENCE=L-EST ID1=19
DEFINE C2L20 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
    SENTENCE=L-EST ID1=20
DEFINE C2L21 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
    SENTENCE=L-EST ID1=21
DEFINE C2L22 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
    SENTENCE=L-EST ID1=22
DEFINE C2L23 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
    SENTENCE=L-EST ID1=23
DEFINE C2L24 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
    SENTENCE=L-EST ID1=24
DEFINE C2V1 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
    SENTENCE=V-EST ID1=1
DEFINE C2V2 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
    SENTENCE=V-EST ID1=2
DEFINE C2V3 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
    SENTENCE=V-EST ID1=3
```

```
DEFINE C2V4 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
    SENTENCE=V-EST ID1=4
DEFINE C2V5 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
    SENTENCE=V-EST ID1=5
DEFINE C2V6 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
    SENTENCE=V-EST ID1=6
DEFINE C2V7 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
    SENTENCE=V-EST ID1=7
DEFINE C2V8 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
    SENTENCE=V-EST ID1=8
DEFINE C2V9 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
    SENTENCE=V-EST ID1=9
DEFINE C2V10 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
    SENTENCE=V-EST ID1=10
DEFINE C2V11 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
    SENTENCE=V-EST ID1=11
DEFINE C2V12 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
    SENTENCE=V-EST ID1=12
DEFINE C2V13 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
    SENTENCE=V-EST ID1=13
DEFINE C2V14 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
    SENTENCE=V-EST ID1=14
DEFINE C2V15 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
    SENTENCE=V-EST ID1=15
DEFINE C2V16 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
    SENTENCE=V-EST ID1=16
DEFINE C2V17 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
    SENTENCE=V-EST ID1=17
DEFINE C2V18 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
    SENTENCE=V-EST ID1=18
DEFINE C2V19 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
    SENTENCE=V-EST ID1=19
DEFINE C2V20 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
    SENTENCE=V-EST ID1=20
DEFINE C2V21 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
    SENTENCE=V-EST ID1=21
DEFINE C2V22 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
    SENTENCE=V-EST ID1=22
DEFINE C2V23 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
    SENTENCE=V-EST ID1=23
DEFINE C2V24 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
    SENTENCE=V-EST ID1=24
DEFINE C4L1 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
    SENTENCE=STR-L-EST ID1=C4 ID2=1
DEFINE C4L2 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
```

```
        SENTENCE=STR-L-EST ID1=C4 ID2=2
    DEFINE C4L3 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
        SENTENCE=STR-L-EST ID1=C4 ID2=3
    DEFINE C5L1 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
        SENTENCE=STR-L-EST ID1=C5 ID2=1
    DEFINE C5L2 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
        SENTENCE=STR-L-EST ID1=C5 ID2=2
    DEFINE C5L3 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
        SENTENCE=STR-L-EST ID1=C5 ID2=3
    DEFINE C4V1 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
        SENTENCE=STR-V-EST ID1=C4 ID2=1
    DEFINE C4V2 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
        SENTENCE=STR-V-EST ID1=C4 ID2=2
    DEFINE C4V3 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
        SENTENCE=STR-V-EST ID1=C4 ID2=3
    DEFINE C5V1 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
        SENTENCE=STR-V-EST ID1=C5 ID2=1
    DEFINE C5V2 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
        SENTENCE=STR-V-EST ID1=C5 ID2=2
    DEFINE C5V3 BLOCK-VAR BLOCK=C2C4C5 VARIABLE=FLOW  &
        SENTENCE=STR-V-EST ID1=C5 ID2=3
C READ PLANT SPECIFICATIONS FROM FILE
F     WRITE(NTERM,*) '+++ PLANT SPECIFICATION FORTRAN STARTED +++'
F     INQUIRE (FILE='ROSMST.DAT', EXIST=THERE)
F     IF (.NOT.THERE) THEN
F       WRITE(NTERM,*) '*** ERROR: ROSMST.DAT DOES NOT EXIST ***'
F       WRITE(NRPT,*) '*** ERROR: ROSMST.DAT WAS NOT FOUND ***'
F       WRITE(NHSTRY,*) '*** ERROR: ROSMST.DAT DID NOT EXIST ***'
F       GO TO 92
F     END IF
F     OPEN (UNIT=63, STATUS='OLD', FILE='ROSMST.DAT')
C   STAB STREAM SPECIFICATIONS
F     READ(63,100) STABF
F     READ(63,100) STABT
F     READ(63,100) STABP
F     READ(63,100) STBZ1
F     READ(63,100) STBZ2
F     READ(63,100) STBZ3
F     READ(63,100) STBZ4
F     READ(63,100) STBZ5
F     READ(63,100) STBZ6
F     READ(63,100) STBZ7
F     READ(63,100) STBZ8
F     READ(63,100) STBZ9
F     READ(63,100) STBZ10
```

```
F     READ(63,100) STBZ11
F     READ(63,100) STBZ12
F     READ(63,100) STBZ13
F     READ(63,100) STBZ14
F     READ(63,100) STBZ15
F     READ(63,100) STBZ16
F     READ(63,100) STBZ17
F     READ(63,100) STBZ18
F     READ(63,100) STBZ19
F     READ(63,100) STBZ20
F     READ(63,100) STBZ21
F     READ(63,100) STBZ22
F     READ(63,100) STBZ23
C   C2REBL STREAM ESTIMATIONS
F     READ(63,100) C2RBF1
F     C2RBF2 = C2RBF1
F     READ(63,100) C2RBT1
F     C2RBT2 = C2RBT1
F     READ(63,100) C2RBP1
F     C2RBP2 = C2RBP1
F     READ(63,100) C2RZ1
F     READ(63,100) C2RZ2
F     READ(63,100) C2RZ3
F     READ(63,100) C2RZ4
F     READ(63,100) C2RZ5
F     READ(63,100) C2RZ6
F     READ(63,100) C2RZ7
F     READ(63,100) C2RZ8
F     READ(63,100) C2RZ9
F     READ(63,100) C2RZ10
F     READ(63,100) C2RZ11
F     READ(63,100) C2RZ12
F     READ(63,100) C2RZ13
F     READ(63,100) C2RZ14
F     READ(63,100) C2RZ15
F     READ(63,100) C2RZ16
F     READ(63,100) C2RZ17
F     READ(63,100) C2RZ18
F     READ(63,100) C2RZ19
F     READ(63,100) C2RZ20
F     READ(63,100) C2RZ21
F     READ(63,100) C2RZ22
F     READ(63,100) C2RZ23
C   E1 HEATER SPECIFICATIONS
F     READ(63,100) E1T
```

```
F     READ(63,100) E1P
C   C1 COLUMN SPECIFICATIONS
F     READ(63,100) C1RR
F     READ(63,100) C1RDV
F     READ(63,100) C1RBFL
F     READ(63,100) C1RBT
F     READ(63,100) C1DP
F     READ(63,100) C1P1
F     READ(63,100) C1P2
C   C2 COLUMN SPECIFICATIONS
F     READ(63,100) C2MD
F     READ(63,100) C2SC
F     READ(63,100) C2DP
F     READ(63,100) C2P1
F     READ(63,100) C2P2
F     READ(63,100) C2IC1
F     READ(63,100) C2IC2
C   C4 COLUMN SPECIFICATIONS
F     READ(63,100) C4QN
C   C5 COLUMN SPECIFICATIONS
F     READ(63,100) C5QN
C   C1 TEMPERATURE PROFILE
F     READ(63,100) C1T1
F     READ(63,100) C1T2
F     READ(63,100) C1T3
F     READ(63,100) C1T4
F     READ(63,100) C1T5
F     READ(63,100) C1T6
F     READ(63,100) C1T7
F     READ(63,100) C1T8
F     READ(63,100) C1T9
F     READ(63,100) C1T10
F     READ(63,100) C1T11
F     READ(63,100) C1T12
F     READ(63,100) C1T13
F     READ(63,100) C1T14
F     READ(63,100) C1T15
F     READ(63,100) C1T16
F     READ(63,100) C1T17
F     READ(63,100) C1T18
F     READ(63,100) C1T19
F     READ(63,100) C1T20
F     READ(63,100) C1T21
F     READ(63,100) C1T22
F     READ(63,100) C1T23
```

```
F     READ(63,100) C1T24
F     READ(63,100) C1T25
F     READ(63,100) C1T26
F     READ(63,100) C1T27
F     READ(63,100) C1T28
F     READ(63,100) C1T29
C   C2 TEMPERATURE PROFILE
F     READ(63,100) C2T1
F     READ(63,100) C2T2
F     READ(63,100) C2T3
F     READ(63,100) C2T4
F     READ(63,100) C2T5
F     READ(63,100) C2T6
F     READ(63,100) C2T7
F     READ(63,100) C2T8
F     READ(63,100) C2T9
F     READ(63,100) C2T10
F     READ(63,100) C2T11
F     READ(63,100) C2T12
F     READ(63,100) C2T13
F     READ(63,100) C2T14
F     READ(63,100) C2T15
F     READ(63,100) C2T16
F     READ(63,100) C2T17
F     READ(63,100) C2T18
F     READ(63,100) C2T19
F     READ(63,100) C2T20
F     READ(63,100) C2T21
F     READ(63,100) C2T22
F     READ(63,100) C2T23
F     READ(63,100) C2T24
C   C4 TEMPERATURE PROFILE
F     READ(63,100) C4T1
F     READ(63,100) C4T2
F     READ(63,100) C4T3
C   C5 TEMPERATURE PROFILE
F     READ(63,100) C5T1
F     READ(63,100) C5T2
F     READ(63,100) C5T3
C   C1 LIQUID AND VAPOUR FLOW PROFILES
F     READ(63,100) C1L1
F     READ(63,100) C1L2
F     READ(63,100) C1L3
F     READ(63,100) C1L4
F     READ(63,100) C1L5
```

```
F     READ(63,100) C1L6
F     READ(63,100) C1L7
F     READ(63,100) C1L8
F     READ(63,100) C1L9
F     READ(63,100) C1L10
F     READ(63,100) C1L11
F     READ(63,100) C1L12
F     READ(63,100) C1L13
F     READ(63,100) C1L14
F     READ(63,100) C1L15
F     READ(63,100) C1L16
F     READ(63,100) C1L17
F     READ(63,100) C1L18
F     READ(63,100) C1L19
F     READ(63,100) C1L20
F     READ(63,100) C1L21
F     READ(63,100) C1L22
F     READ(63,100) C1L23
F     READ(63,100) C1L24
F     READ(63,100) C1L25
F     READ(63,100) C1L26
F     READ(63,100) C1L27
F     READ(63,100) C1L28
F     READ(63,100) C1L29
F     READ(63,100) C1V1
F     READ(63,100) C1V2
F     READ(63,100) C1V3
F     READ(63,100) C1V4
F     READ(63,100) C1V5
F     READ(63,100) C1V6
F     READ(63,100) C1V7
F     READ(63,100) C1V8
F     READ(63,100) C1V9
F     READ(63,100) C1V10
F     READ(63,100) C1V11
F     READ(63,100) C1V12
F     READ(63,100) C1V13
F     READ(63,100) C1V14
F     READ(63,100) C1V15
F     READ(63,100) C1V16
F     READ(63,100) C1V17
F     READ(63,100) C1V18
F     READ(63,100) C1V19
F     READ(63,100) C1V20
F     READ(63,100) C1V21
```

```
F     READ(63,100) C1V22
F     READ(63,100) C1V23
F     READ(63,100) C1V24
F     READ(63,100) C1V25
F     READ(63,100) C1V26
F     READ(63,100) C1V27
F     READ(63,100) C1V28
F     READ(63,100) C1V29
C   C2 LIQUID AND VAPOUR FLOW PROFILES
F     READ(63,100) C2L1
F     READ(63,100) C2L2
F     READ(63,100) C2L3
F     READ(63,100) C2L4
F     READ(63,100) C2L5
F     READ(63,100) C2L6
F     READ(63,100) C2L7
F     READ(63,100) C2L8
F     READ(63,100) C2L9
F     READ(63,100) C2L10
F     READ(63,100) C2L11
F     READ(63,100) C2L12
F     READ(63,100) C2L13
F     READ(63,100) C2L14
F     READ(63,100) C2L15
F     READ(63,100) C2L16
F     READ(63,100) C2L17
F     READ(63,100) C2L18
F     READ(63,100) C2L19
F     READ(63,100) C2L20
F     READ(63,100) C2L21
F     READ(63,100) C2L22
F     READ(63,100) C2L23
F     READ(63,100) C2L24
F     READ(63,100) C2V1
F     READ(63,100) C2V2
F     READ(63,100) C2V3
F     READ(63,100) C2V4
F     READ(63,100) C2V5
F     READ(63,100) C2V6
F     READ(63,100) C2V7
F     READ(63,100) C2V8
F     READ(63,100) C2V9
F     READ(63,100) C2V10
F     READ(63,100) C2V11
F     READ(63,100) C2V12
```

```
F     READ(63,100) C2V13
F     READ(63,100) C2V14
F     READ(63,100) C2V15
F     READ(63,100) C2V16
F     READ(63,100) C2V17
F     READ(63,100) C2V18
F     READ(63,100) C2V19
F     READ(63,100) C2V20
F     READ(63,100) C2V21
F     READ(63,100) C2V22
F     READ(63,100) C2V23
F     READ(63,100) C2V24
C   C4 LIQUID AND VAPOUR FLOW PROFILES
F     READ(63,100) C4L1
F     READ(63,100) C4L2
F     READ(63,100) C4L3
F     READ(63,100) C4V1
F     READ(63,100) C4V2
F     READ(63,100) C4V3
F     READ(63,100) C5L1
F     READ(63,100) C5L2
F     READ(63,100) C5L3
F     READ(63,100) C5V1
F     READ(63,100) C5V2
F     READ(63,100) C5V3
F     CLOSE (UNIT=63)
F     WRITE(NTERM,*) 'ROSMST.DAT READ SUCCESSFULLY'
F  92 WRITE(NTERM,*) '+++ PLANT SPECIFICATION FORTRAN COMPLETED +++'
C FORMAT STATEMENTS
F 100 FORMAT(15X, E15.8)
    EXECUTE FIRST

CONV-OPTIONS
    PARAM TRACEOPT=GRADUAL CHECKSEQ=YES

CONVERGENCE C2REBL WEGSTEIN
    DESCRIPTION "C2 REBOILER TEAR STREAM"
    BLOCK-OPTION RESTART=YES
    TEAR C2REBL

REPORT NOINSERT NOADA NOSENSITIVIT NOPROPERTIES NOCOSTBLOCK  &
        NOUNITS NOUTILITIES NOECONOMIC

BLOCK-REPORT NOSORT NOTOTBAL NOINPUT
```

```
STREAM-REPOR WIDE NOMOLEFLOW MOLEFRAC NOVOLFRAC NOATTR-DESC  &
        NOCOMP-ATTR NOSUBS-ATTR PROPERTIES=TBP-10 TBP-90 STDVOL  &
        V-STDVOL VOL RVP TBUB EXCL-STREAMS=C2BOT C2RBFD

FLOWSHEET-RE NODESCRIPTIO NOTOTBAL NOCOMPBAL NOSEQUENCE  &
        NOFORTRAN NODESIGN-SPE NOCONSTRAINT NOOPTIMIZATI  &
        NOTRANSFER
```

F.2 randdata.f

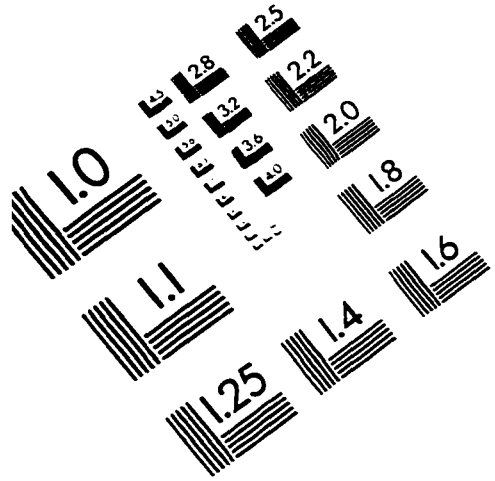
The following FORTRAN77, `randdata.f`, adds normally distributed random noise to the measured plant data. This program requires the NAG Library FORTRAN Routines.

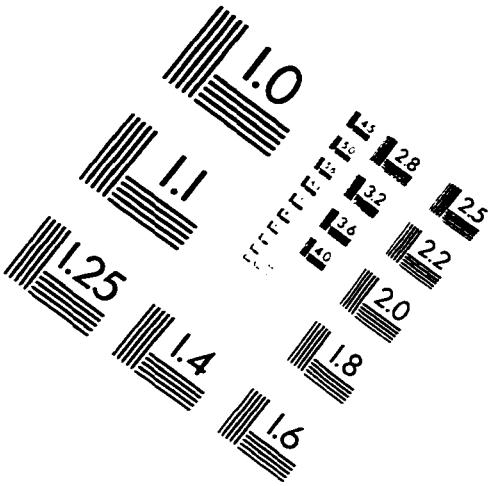
```
C RANDDATA.F - ROUTINE FOR ADDING NORMALLY DISTRIBUTED NOISE TO DATA
C
C NOTE: This routine uses NAG Library Routines, Compile on Cape with the
C       command line: f77 randdata.f -lnag
C
C DECLARATIONS:
      INTEGER M, NPTS
      DOUBLE PRECISION V(28), RV(28), SV(28)
      CHARACTER *13 VLAB(28)
      LOGICAL THERE
C   External Functions:
      DOUBLE PRECISION G05DDF
      EXTERNAL G05DDF
C   External Subroutines:
      EXTERNAL G05CBF
C   Set Up Variables:
      NPTS=28
      WRITE (*,*) '>>>RANDDATA CALLED'
C READ IN DATA:
C   Read in PLVARS.DAT
      INQUIRE (FILE='PLVARS.DAT', EXIST=THERE)
      IF (.NOT.THERE) THEN
        GO TO 915
      ENDIF
      OPEN (UNIT=50, STATUS='OLD', ERR=910, FILE='PLVARS.DAT')
      REWIND (UNIT=50)
      DO 20 M=1,NPTS
        READ (50,100,ERR=900,END=905) VLAB(M), V(M)
```

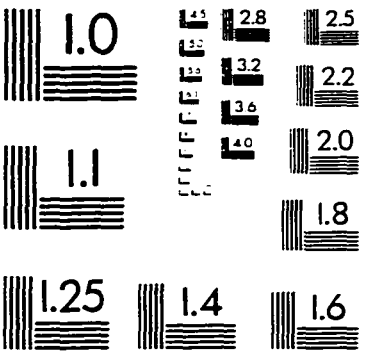
```
   20 CONTINUE
      CLOSE (UNIT=50)
C    Read in RVARNC.DAT
      INQUIRE (FILE='RVARNC.DAT', EXIST=THERE)
      IF (.NOT.THERE) THEN
        GO TO 916
      ENDIF
      OPEN (UNIT=50, STATUS='OLD', ERR=910, FILE='RVARNC.DAT')
      REWIND (UNIT=50)
      DO 25 M=1,NPTS
        READ (50,110,ERR=902,END=906) SV(M)
   25 CONTINUE
      CLOSE (UNIT=50)
C MAIN PROGRAM:
      CALL G05CBF(0)
      DO 30 M=1,NPTS
        RV(M) = V(M) + G05DDF(0, SV(M))
   30 CONTINUE
C WRITE OUT DATA:
      INQUIRE (FILE='IMEAS.DAT', EXIST=THERE)
      IF (THERE) THEN
        GO TO 920
      ENDIF
      OPEN (UNIT=51, STATUS='NEW', ERR=910, FILE='IMEAS.DAT')
      WRITE(*,107) 'VARIABLE', 'OLD VALUE', 'VALUE WITH NOISE'
      DO 40 M=1,NPTS
        WRITE(51,100,ERR=901) VLAB(M), RV(M)
        WRITE(*,105) VLAB(M), V(M), RV(M)
   40 CONTINUE
      CLOSE (UNIT=51)
      WRITE (*,*) '>>>RANDDATA - NOISY VARIABLES WRITTEN SUCCESSFULLY'
      GO TO 1000
C FORMAT STATEMENTS:
  100 FORMAT(A13, 2X, E15.8)
  105 FORMAT(A13, 2X, E15.8, 2X, E15.8)
  107 FORMAT(A13, 2X, A15, 2X, A17)
  110 FORMAT(15X, E15.8)
C ERROR STATEMENTS:
  900 WRITE(*,*) 'RANDATA: Error reading PLVARS.DAT line:', M
      GO TO 1000
  901 WRITE(*,*) 'RANDATA: Error writing IMEAS.DAT line:', M
      GO TO 1000
  902 WRITE(*,*) 'RANDATA: Error reading RVARNC.DAT line:', M
      GO TO 1000
  905 WRITE(*,*) 'RANDATA: EOF in PLVARS.DAT reading line: ', M
```

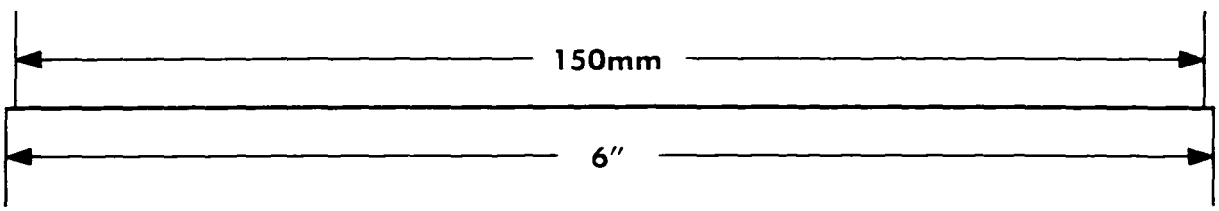
```
      GO TO 1000
  906 WRITE(*,*) 'RANDATA: EOF in RVARNC.DAT reading line: ', M
      GO TO 1000
  910 WRITE(*,*) 'RANDATA: Error opening file'
      GO TO 1000
  915 WRITE(*,*) 'RANDATA: PLVARS.DAT does not exist'
      GO TO 1000
  916 WRITE(*,*) 'RANDATA: RVARNC.DAT does not exist'
      GO TO 1000
  920 WRITE(*,*) 'RANDATA: IMEAS.DAT already exists'
      GO TO 1000
 1000 WRITE(*,*)
      END
```

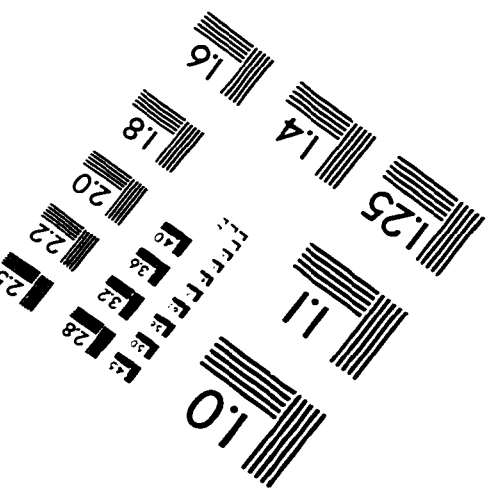
IMAGE EVALUATION
TEST TARGET (QA-3)

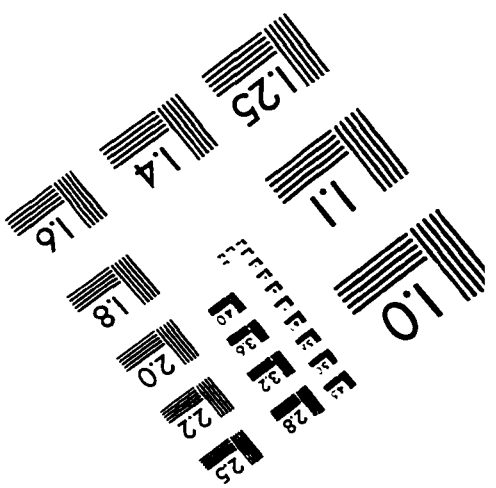












APPLIED IMAGE, Inc
1653 East Main Street
Rochester, NY 14609 USA
Phone: 716/482-0300
Fax: 716/288-5989

© 1993, Applied Image, Inc., All Rights Reserved