

# Dynamic Optimization of the Tennessee Eastman Process Using the OptControlCentre

Tobias Jockenhövel\*, Lorenz T. Biegler\*\* and Andreas Wächter\*\*\*

\* Institute for Energy Engineering, Technical University of Berlin,  
Marchstrasse 18, 10587 Berlin, Germany, Tobias@Jockenhoevel.com

\*\* Department of Chemical Engineering, Carnegie Mellon University,  
Pittsburgh, PA 15213, USA, lb01@andrew.cmu.edu

\*\*\* IBM T.J. Watson Research Center,  
P.O. Box 218, Yorktown Heights, NY 10598, USA, andreasw@watson.ibm.com

This study focuses on the performance of large-scale nonlinear programming (NLP) solvers for the dynamic optimization in real-time of large processes. The MATLAB-based OptControlCentre (OCC) is coupled with large-scale optimization tools and developed for on-line real-time dynamic optimization. To demonstrate these new developments, we consider the on-line, real-time dynamic optimization of the Tennessee Eastman (TE) Challenge Process in a nonlinear model predictive control (NMPC) framework. The example captures the behavior of a typical industrial process and consists of a two phase reactor, where an exothermic reaction occurs, along with a flash, a stripper, a compressor and a mixer. The process is nonlinear and open loop unstable; without control it reaches shutdown limits within an hour, even for very small disturbances. The system is represented through a first principles model with about 200 differential algebraic equations (DAEs). As a result, the NMPC formulation of this system presents some interesting features for dynamic optimization approaches. This study compares two state-of-the-art NLP solvers, SNOPT and IPOPT, for dynamic optimization on a number of challenging control scenarios, and illustrates some of the advantages of IPOPT for dynamic optimization.

## Keywords

On-line optimization, SQP, NLP solvers, real-time optimization, RTO, NMPC

## 1 Introduction

With the development of superior model-based control strategies and more efficient numerical solvers, dynamic optimization has become a viable tool for process systems engineering. A number of modeling systems, including commercial tools such as gProms (Process Systems Enterprises Limited, 2002) and Aspen Custom Modeler (Aspen Technology, 2002), have also increased the awareness and activity in dynamic simulation and optimization based on first principles models. As a result, very large optimization models can be formulated that represent complex process systems. However, a further challenge is the execution of dynamic optimization models in real-time for on-line purposes, including system identification and data reconciliation, model predictive control and real-time optimization. These require a combination of efficient and stable dynamic optimization problem formulations as well as fast large-scale nonlinear

programming tools. Over the past five years, a number of advances have been made on these fronts.

In particular, a number of key concepts have been developed for the efficient problem formulation for real-time optimization (RTO) and nonlinear model predictive control (NMPC). Here, NMPC formulations usually rely on the prediction of a dynamic model over moving time horizons. Process state and output variables are predicted over an output horizon and manipulated variables are determined on a (possibly shorter) input horizon. Formulation of these moving time horizon problems is dictated by stability properties of the controller. As a result, MPC formulations with infinite time horizons, endpoint constraints and dual mode controllers (infinite horizon problems composed of finite horizon problems followed by linear feedback controllers) (see Mayne et al. (2000) for a review) have been considered. In addition, the representation and solution of the differential-algebraic equation (DAE) model is a key part of this formulation and DAEs for NMPC have been incorporated using sequential, single shooting formulations (Vassiliadis, Sargent & Pantelides, 1994) (with the DAEs solved in an inner loop and manipulated variables determined in an outer optimization loop), simultaneous formulations in the state and manipulated variables (Betts, 2001; Cervantes et al., 2000), and in-between formulations which rely on multiple shooting (Diehl et al., 2002; Leineweber et al., 1997). The relative merits of these strategies are discussed in (Biegler, Cervantes & Wächter, 2002; Groetschel, Krumke & Rambau, 2001) and there are active research efforts in the refinement of all of these strategies.

In the past decade, nonlinear programming tools have advanced rapidly for RTO and NMPC. Currently, it is not uncommon to formulate and solve NLP problems with  $10^5$ - $10^6$  variables and constraints. Moreover, the availability of second derivatives in some modeling platforms (e.g., AMPL) allows more sophisticated methods to be used. Here we classify these methods into the following categories: *active set vs. barrier methods* to handle bounds and inequality constraints, *providing second order information* and exploiting problem structure through full space or reduced space methods and *line search and trust region* methods to enforce global convergence. These issues will be discussed in Section 2 related to the IPOPT (Wächter, 2002) and SNOPT (Gill, Murray & Saunders, 1998) solvers.

Finally, RTO and NMPC strategies require a flexible and powerful modeling and software environment that allows evaluation of these features for dynamic optimization strategies on large-scale processes. In addition to interfaces for flexible model development and optimization problem formulation, this environment includes automatic differentiation facilities to construct Jacobian and Hessian information from the DAE model, a rich choice of discretization schemes for DAE models and graphical utilities to observe and assess the performance of the optimization strategy.

In this study, we discuss the above features and combine them into a novel dynamic optimization package, called OCC (OptControlCentre). OCC implements a simultaneous optimization approach for RTO and NMPC. In the next section we describe the primary structure of OCC and also discuss the SNOPT and IPOPT algorithms, which are incorporated in OCC. As a case study that illustrates the capabilities of OCC and the performance of the dynamic optimization, we consider the Tennessee Eastman challenge problem. The DAE model for this case study is presented in Section 3 and a number of control scenarios are presented in Section 4. Concluding remarks and directions for future work are given in Section 5.

## 2 Dynamic Optimization Platform

The dynamic optimization approach described in this section consists of two parts. First, a software platform is described for the formulation, validation and interpretation of the dynamic optimization problem. This platform is built on MATLAB and Maple tools. Second, powerful large-scale NLP solvers are required in order to solve the discretized dynamic optimization problem within the constraints of on-line operation.

### 2.1 OptControlCentre

The software platform OptControlCentre (OCC) was developed for the implementation of dynamic online optimization. Built on a MATLAB framework and using Maple for the generation of exact derivatives, OCC (Jockenhövel, 2002, 2003) allows its users to perform dynamic optimization of complex problems with a powerful graphical user interface. Because OCC is based on MATLAB, it runs on major computing platforms including Windows, Linux or UNIX. The structure of OCC / OCOMA is shown in Figure 1. The continuous process model is defined as a set of DAEs using a relatively simple syntax in the first stage of OCC. In the second stage, the Maple-based code generator OCOMA (Bausa, 2000; Jockenhövel, 2003) transforms the continuous DAE optimization problem to a NLP problem through a full discretization of state and control variables. Various discretization schemes are available in OCOMA including:

- Implicit Euler,
- Trapezoidal Rule
- Central Differences (2<sup>nd</sup> and 4<sup>th</sup> order),
- Backward Differential Formulae – BDF (2<sup>nd</sup> and 3<sup>rd</sup> order)
- Radau Collocation on Finite Elements (2<sup>nd</sup> and 3<sup>rd</sup> order).

OCOMA then automatically writes the FORTRAN subroutines that contain the NLP problem, including symbolic (exact) first and second derivatives (Hessian or Hessian-Vector-Products for IPOPT). In the third stage, these codes are then linked with large-scale NLP solvers which perform the dynamic optimization. The OptControlCentre allows both off-line and on-line optimization, and the optimization results can be visualized using the graphical interface built with MATLAB tools.

Both dynamic and stationary nonlinear optimization can be conducted with the simultaneous optimization of both control profiles and control parameters. OCC can also be used for dynamic system identification. Moreover, the OptControlCentre can conduct fully automatic optimizations of dynamic systems in real-time. Here the state of the plant can be updated by a nonlinear dynamic system identification, which provides estimation values of unmeasured system variables. System identification (with or without noise) is either done for the entire dynamic system or for various subsystems in parallel, which is much faster. This step is then followed by a nonlinear dynamic optimization which provides updated optimal profiles of the control variables. The course of the online optimization can be supervised by displaying the system and through appropriate problem formulation; the user can easily conduct studies on scheduling, multiple objectives and fault detection.

## 2.2 Solving the NLP Problem

For NMPC we consider NLP solvers based on successive quadratic programming (SQP) as these have been shown to require the fewest function evaluations to solve NLPs (Schittkowski, 1987; Binder, et al., 2001). SQP methods apply the equivalent of Newton steps to the optimality conditions of the nonlinear programming problem and this leads to a fast rate of convergence. For large-scale problems we consider two NLP solvers in this work, SNOPT (Gill, Murray & Saunders, 1998) and IPOPT (Wächter, 2002), and several issues are discussed below to justify this choice.

Consider the following general form of the NLP:

$$\min \mathbf{f}(x) \tag{1}$$

$$\text{s.t. } h(x) = 0 \tag{2}$$

$$g(x) = 0 \tag{3}$$

Here (1), (2) and (3) represent the objective function, equality constraints and inequality constraints of the NLP, respectively. Bounds on variables are a special case of (3). At a stationary point,  $x^*$ , the first order KKT conditions for this problem are given by:

$$\tilde{\mathbf{N}}\mathbf{f}(x^*) + A(x^*) \mathbf{I} + C(x^*) \mathbf{v} = 0 \tag{4}$$

$$h(x^*) = 0 \tag{5}$$

$$g(x^*) + s = 0 \tag{6}$$

$$\mathbf{S} \mathbf{V} e = 0 \tag{7}$$

$$(s, \mathbf{v}) = 0 \tag{8}$$

where  $e = [1, 1, \dots, 1]^T$

$\mathbf{I}$ : the multipliers of the equalities

$\mathbf{v}$ : the multipliers of the inequalities

$\mathbf{S} = \text{diag}\{s\}$

$\mathbf{V} = \text{diag}\{v\}$

$A(x) = \tilde{\mathbf{N}}h(x)$

$C(x) = \tilde{\mathbf{N}}g(x)$

SQP methods find solutions that satisfy (4)-(8) by generating Newton-like search directions at iteration  $k$ . In general, one can classify SQP methods by the following categories:

- *active set vs. barrier methods* to handle bounds and inequality constraints in generating search directions. With active set methods, a combinatorial search is required, to determine which of the linearized inequalities should strictly satisfied. Barrier methods, on the other hand, relax equation (7) by a parameter  $\mathbf{m}$  and a series of subproblems is solved where  $\mathbf{m}$  is forced to zero.
- *second order information* can be provided exactly, or approximated by a number of quasi-Newton methods. Moreover, the problem structure can be exploited for Newton-like steps that arise from (4)-(8).

- *line search* and *trust region* methods enforce global convergence of the SQP iterations.

The two SQP codes, SNOPT and IPOPT, that are interfaced to OCC have the following characteristics. SNOPT solves the QP subproblem using an active set strategy. It applies a line search algorithm based on an *augmented Lagrangian* function, a limited memory BFGS update to *approximate* second order information, and an efficient reduced space decomposition to improve the solution of the QP subproblem. Moreover, one of the key features of SNOPT is the capability to distinguish between nonlinear and linear parts of the model, specified by the user. To exploit this feature, OCOMA can automatically conduct a linearity analysis of a given model. For the TE Process, 132 nonlinear and 50 linear variables have been identified. The majority of the model (72.25%) is therefore nonlinear. Finally, SNOPT further incorporates a warm start feature which allows reuse of matrix factorizations and active constraints from an earlier optimization. In this work, however, these features did not lead to a significant reduction of the number of iterations or the computational time.

In contrast, IPOPT is a barrier method that proceeds by solving a relaxed version of (4)-(8), to relaxed tolerances, for decreasing values of  $\mu$ . Here the algorithm applies a Newton method to this system in the full space, using a sparse indefinite symmetric linear solver. Exact second derivatives (provided by Maple in OCOMA) are calculated and safeguards are added to ensure certain descent properties (Wächter, 2002). To ensure global convergence, a *filter line search* method is used and second order corrections are also used to maintain a fast convergence rate close to the solution. More information on IPOPT can be found in (Biegler, Cervantes & Wächter, 2002; Wächter, 2002; Wächter and Biegler, 2002).

Finally, because our dynamic optimization problems are solved repeatedly, a warm start feature has also been implemented in IPOPT. This feature allows the specification of  $\mathbf{m}$  at a low restart value (e.g.  $\mathbf{m} = 10^{-4}$ ) and it allows reuse of Lagrange multiplier values from a previous optimization. As seen in Table 1, this warm start feature reduces the number of iterations and CPU time significantly. The effect of the warm start feature is shown for initial barrier parameter values of  $\mu=10^{-1}$  to  $\mu=10^{-6}$  where  $\mu=10^{-1}$  is equal to its cold start value. As can be seen for the base case optimization in Table 1, for initial barrier parameters set too low, the optimizations need more iterations again. As a result, optimizations with IPOPT in this study have been chosen to use  $\mu=10^{-4}$  as the initial barrier parameter.

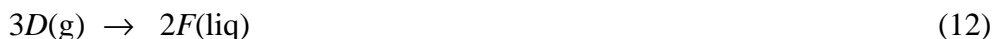
### 3 The Tennessee Eastman Process

The Tennessee Eastman Process (TE process) is a plant-wide process control problem proposed by Downs & Vogel (1993) as a challenge test problem for a number of control related topics, including multivariable controller design, optimization, predictive control, estimation/adaptive control, nonlinear control, process diagnostics, and education. Since the publication of the TE process example in 1993, over 60 studies have used this case study for plant-wide control, statistical process monitoring, sensor fault detection, and identification of data-driven neural network models. A review of plant wide control strategies applied to the TE process is given in (Larsson & Skogestad, 2000) and self-optimizing control structures are described in Larsson, Hestetun, Hovland & Skogestad (2001). In addition, a number of studies deal with on-line process optimization of the TE process. In several studies, NMPC strategies have been developed

for reduced models of the TE process (Ricker & Lee, 1995a, 1995b) or its components (Albuquerque, et al., 1999; Zheng & Zhang, 2001), with the goal of demonstrating properties of the NMPC problem formulation or performance of the NMPC algorithm. In addition, steady state on-line optimization of the TE process was considered in (Duvall & Riggs, 2000; Ricker, 1995) and the coordination of different components of the RTO steps was described in Yan & Ricker (1997). In this study we consider the performance of large-scale NLP solvers for the dynamic on-line optimization of large processes. In particular, we focus on the dynamic optimization of the *entire* detailed Tennessee Eastman process with almost 200 DAE model equations. As described in Binder, et al. (2001), NMPC requires the efficient solution of dynamic optimization problems for control optimization. In Section 4 we demonstrate that our approach is effective for large scale optimization problems of this type.

### 3.1 Process Description

The TE process produces two products ( $G$  and  $H$ )<sup>1</sup> from four reactants ( $A$ ,  $C$ ,  $D$  and  $E$ ). A further inert trace component ( $B$ ) and one byproduct ( $F$ ) are present. The reactions are:



These reactions are irreversible and exothermic with rates that depend on temperature and on the reactor gas phase concentration of the reactants. Figure 2 shows the flowsheet of the TE process taken from Downs and Vogel. The process consists of a continuous stirred tank reactor, a condenser, a flash drum and a stripper. Here the reactants  $A$ ,  $C$ ,  $D$  and  $E$  are partly converted into products  $G$ ,  $H$  and the (undesired) byproduct  $F$ . The heat of reaction is removed by an internal heat exchanger using water as cooling medium. The products and unconverted reactants leave the reactor as vapor.

This vapor is partly liquefied in the condenser and the two phases are separated in the flash drum. A purge is necessary to prevent the inert component  $B$  from building up in the recycle stream. The recycle is then fed back to the mixing zone, using a compressor to compensate for pressure losses in the reactor, condenser and flash. The liquid bottom stream of the flash is pumped into a stripper in order to obtain the desired product purity. Here the remaining impurities – mainly unconverted reactants – are vaporized and fed back into the mixing zone. Finally, the stripper bottom stream consists of a mixture of the products  $G$  and  $H$ , containing only traces of impurities. The original process has 12 manipulated variables and 41 measurements. Further operating constraints for temperatures, pressures and liquid levels for the unit operations are given in Table 3.

### 3.2 Process Model of the Tennessee Eastman Process

The process model was originally developed as an undergraduate project (Feller, 1997) and was adapted here to be used with OCC. This model is based on the description and FORTRAN code

---

<sup>1</sup> The real components are not published by Downs and Vogel in order to protect the proprietary nature of the process. Because of this reason also the kinetics and the process and operating conditions are altered.

of Downs & Vogel (1993) and Ricker and Lee (1995a, 1995b), but in our work their model is extended in a number of ways. To provide a more realistic model we have added energy balances for the reactor, the product separator, the stripper and the mixing zone. These are added in order to use the manipulated variables given by Downs and Vogel (1993) directly by the predictive controller instead of the manipulated variables that Ricker and Lee (1995a, 1995b) suggest. With this and other modifications described in the following, we have 30 differential states, 160 algebraic states and 11 manipulated variables (Table 2), compared to the model in Ricker and Lee (1995a, 1995b) which has 26 states and 10 manipulated variables.

Because of this addition, our nonlinear predictive controller now needs to control an open loop unstable system, which is more challenging than controlling a system already stabilized by PI-controllers. For the purposes of this optimization study, we believe that the ability to handle ill-conditioning due to instabilities in NMPC to be an important characteristic in the choice of NLP solvers and dynamic optimization approaches.

Analyzer delay is not included in the problem and no noises are given on the measurements. Adding the delay would be essential for the evaluation of controller performance. However, it is not included here as it does not add much for assessing the performance of a dynamic optimization algorithm. Finally, as the focus of this study is on the performance of the dynamic optimization strategy, noise is not added to the outputs or to the inputs in this model. The model is summarized in Figure 3 - Figure 6 with all variables defined in the nomenclature section. The reader is also referred to Jockenhövel (2002, 2003) for a detailed description of the model.

### **3.2.1 Mixing Zone**

Within the mixing zone all feed streams and the recycle stream are mixed and fed into the reactor. All components are in vapor phase and thus the model reduces to mass and energy balances plus two algebraic equations for mixing zone pressure and composition. The mixing zone model is summarized in Figure 3.

### **3.2.2 Reactor**

The reactor contains liquid and vapor phases, which are assumed to be in equilibrium. Further, four exothermic reactions take place (9)-(12) and excess heat is removed by cooling water. The feed and the product streams of the reactor are vapor phase. Influences on the agitator within the reactor are neglected as the agitator is excluded from the model. The reactor model is summarized in Figure 4.

### **3.2.3 Product Separator**

The separator model in Figure 5 is similar to the reactor model, without terms for the chemical reactions. The vapor-liquid equilibrium is described by a constant activity coefficient for each (condensable) component.

### **3.2.4 Compressor and Purge**

We also follow the approach by Ricker and Lee (1995b) by introducing the recycle stream as an independent variable. This avoids modeling the compressor in detail and the stream flow rate no longer depends on the valve position in the compressor recycle. Temperature changes due to the compressor work are taken into account by the equation shown in Figure 5. Similar to the

compressor simplification, the purge stream is a simple algebraic variable. These simplifications do not alter the nature of the optimization problem as the compressor and purge model equations are only algebraic equations. The actual setpoints for the valves are recalculated from the optimization results.

### 3.2.5 Stripper

Because there is little information available about the stripper, this unit is characterized by a simple split fraction model as proposed in Ricker and Lee (1995b). Here a similar model is used, but an energy balance is added and split fractions ( $\Phi_i$ ) are now modeled as third degree polynomials in temperature. The coefficients of these polynomials fit the model of the TE process around the base case. The pressures in the stripper and the mixing zone are assumed to be equal. The heating medium is saturated steam, which condenses completely at a constant temperature. The enthalpy of the steam has been chosen to fit the steam flux given in Downs and Vogel (1993) for the base case with the heat duty of the stripper at base case. The stripper process model is summarized in Figure 6.

## 3.3 Dimension of the Optimization Problem

The OCOMA analysis for the entire TE model gives a system with 30 differential equations and 149 algebraic equations with 11 control variables. The actual numbers of equations are higher than the sum of all model equations presented above. This is due to a reformulation for all denominators, square roots, logarithms, etc. to make the model better behaved. The optimization model thus contains more linear elements at the expense of (slightly) increasing the problem size.

The optimization model seen by the NLP solver is also much larger due to the full discretization of state and control variables. Along with the variable bounds, the TE model, discretized over 60 difference points with the second order Backward Differential Formulae (BDF) leads to a large-scale optimization problem with 11400 optimization variables, 10740 constraints, 50550 nonzero entries in the Jacobian and 15180 nonzero entries in the Hessian, as shown in Table 5. There are 660 degrees of freedom in the resulting NLP problem. Symbolic (exact) second derivatives are used for IPOPT, whereas SNOPT uses the BFGS approach.

## 3.4 Objective Formulation

The code generator OCOMA offers three ways to formulate the objective function:

- by an algebraic expression maximized or minimized at final time,
- by an differential equation which is integrated and its integral is either maximized or minimized at final time  $t_f$ ,
- by an algebraic expression which is evaluated at all difference points and its sum is minimized or maximized (discrete objective mode)

In this work, the discrete objective mode was used and the objective is given by:

$$\min_{u(t)} J(t, u, y) = \sum_{i=1}^{NINT} \left[ (y_i - y_{i,sp})^T Q_y (y_i - y_{i,sp}) + (u_i - u_{i,r})^T Q_u (u_i - u_{i,r}) \right] \quad (13)$$

with NINT as the total number of difference points.

The objective was formulated as a least squares control objective with the 11 manipulated variables as inputs (stream flowrates) and the product concentrations of the components  $G$  and  $H$ , as well as the purge concentrations of  $B$  as output variables. As these are the only relevant outputs from the process all of the other output variables within the process are allowed to vary between their bounds; additional setpoints are not imposed on them. By posing the objective function using product measurements as the only output variables, we believe that this leads to a less restrictive formulation of the optimization problem. The entries were divided by their base case values taken from Downs and Vogel (1993) and the manipulated variable terms multiplied by the factor 0.1. Figure 7 shows the formulation of the objective in the model code.

This controller objective does not provide zero steady-state offset and is usually not considered to be good control practice. It can be forced to work in this idealized situation (no model mismatch) by using manipulated value reference values that exactly result in achieving the set points. However, these values would generally not be known in practice. Because of this practical difficulty we do not try to eliminate steady state offsets. Moreover, this does not change the main point of the paper, which is NLP performance.

An economic objective could also have been considered. However, here we want to follow the NMPC case studies, guidelines and problem formulation outlined in the original Downs and Vogel study for testing our optimization algorithms.

### 3.5 Modified Base Case

Model differences to the original Downs & Vogel model (in particular the added energy balances) resulted in a slight different steady state with temperatures and pressures found to be at their upper bounds. However, reactor temperature and reactor pressure are key state variables due to the unstable nature of the process. To ensure smooth operation close to the setpoint, extra terms were added to regularize the objective in the form of:

$$\begin{aligned} & ( 393.55 - Tr )^2 / 393.55 + \\ & ( 353.25 - Ts )^2 / 353.25 + \\ & ( 338.85 - Tstr )^2 / 338.85 + \\ & ( 2.805 - pr\_MPa )^2 / 2.805 \end{aligned} \tag{14}$$

To find a valid steady state of the model, the base case objective as stated in Figure 7 together with (14) was run with IPOPT over 350 times, resulting clearly to a new steady-state with a slightly higher objective value, see upper plot in Figure 8. With an initial choice of sampling time at 600 seconds, these 350 optimizations equal an optimization/simulation time of over 58 hours. The reactor pressure still tends to be higher than stated in the objective; however, the values are still quite reasonable, as shown in Figure 8. A higher reactor pressure ensures the desired production rate, an observation also made during steady-state optimizations (Ricker, 1995). The modified base case is used in all subsequent optimization studies.

## 4 Dynamic Online-Optimization of the TE problem

Downs and Vogel suggested a set of setpoint changes for case studies, as well as other disturbance scenarios (Downs and Vogel, 1993). Three of the provided setpoint change studies are shown in Table 6 and are considered here. Each setpoint change study should be simulated at least over 24 – 48 hours to see the full results on the process. All case studies are optimized over 1000 cycles using IPOPT with a sampling time of 100 seconds. After 300 cycles, OCC automatically switched from the base case objective to the new setpoint. The online optimization was executed on a Pentium III, 1GHz processor with 512 MB RAM. All results in this study are shown over a transient of 27 hours. The profiles of controls and states exhibit no changes after this time.

The online optimization was run with the following approach. The state and the control variables of the DAE system were fully discretized with the BDF method (second order) over 60 difference points. The sample time was set to 100 seconds. Initially, we selected a 10 minute sampling time. This value was used in previous studies (see Albuquerque et al., 1999) that dealt only with the reactor model. For the overall process, however, this longer sampling time, coupled with the instability, led to oscillations in some of the input profiles (Jockenhövel, 2003). On the other hand, because IPOPT requires far less computational time, a shorter sampling time of 100 seconds could be used here.

The input and output horizon is set to 5 hours; that means each optimization calculates updated control values for the next 5 hours and predicts the system variables over the same time horizon. The optimal control trajectories are applied directly to the process. This output horizon is too short for the process, which takes much longer to achieve steady state. However, unlike MPC with linear models, NMPC implementations do not have the luxury of choosing long output horizons due to computational purposes. This is exacerbated by the unstable nature of the process as well. Nevertheless, the short output horizon still leads to an NMPC controller that performs very well, particularly in the long term. However, in an actual implementation of NMPC, the unstable modes would be controlled with a fast PI controller and one would not expect to use a fast sampling time. Integrating action in our NLP model could also deal with steady-state offset. This leads to additional linear equations that would slightly affect the linear algebra but would not the overall performance of the optimization code.

Various attempts have been made to use different time horizons. However, the performance was found to be insufficient for online optimization. The authors believe this is due to the unstable nature of the model. Normally, the usual online optimization with the OptControlCentre would include a Simulink model as the actual plant and an additional dynamic system identification step. These two steps are not included in this work as the focus lies on the performance of the dynamic optimization. Data reconciliation with state estimation will also be considered in future work. Instead, the model and actual process are assumed identical so that a separate simulation plant model and a system identification step are not required. Also, as mentioned above, control related issues such as disturbances, analyzer delay and model mismatch are not considered in this study.

#### **4.1 Performance Comparison with SNOPT**

The performance of the active set SQP solver SNOPT was found to be insufficient for the direct control of the TE problem. The required solution time of SNOPT usually exceeded the sample times of 100 or even 600 seconds on a Pentium III, 1GHz processor with 512 MB RAM. For the

modified base case, an increasing number of iterations are needed and the calculation times are as high as 7000 CPU s and 600 iterations. The feasibility and optimality tolerances are set to  $10^{-5}$ , the “Hessian Limited memory” option is used as the problem is very large. The “Hessian updates” option has been left to its default value of 20, which means after 20 BFGS updates have been carried out, all but the diagonal elements of the accumulated updates are discarded and the updating process starts again. The process operating constraints from Table 3 are treated as variable bounds; no other variable bounds are set. Several other SNOPT options were tried as well, but with no improvements in performance.

Although optimal solutions could be found in the first five cycles, a new iteration limit of 100 iterations is set to avoid excessive computing times. With this new limit the base case optimization was successful only twice but subsequently failed to converge. Later on, SNOPT was able to solve the problem only three times but with unacceptable computational times over 1000 CPU s. As a result, all of the following studies are calculated using IPOPT with a sampling time of 100 seconds.

## 4.2 Production Rate Change Study

The production rate change requires an overall downscale of the entire process. After 300 optimizations with the modified base case objective formulation, OCC switches automatically to a new objective formulation in which the values for the stream flow rates in the objective have been multiplied with the factor 0.85. Temperatures and pressures in the objective remain unchanged. Scaling both control reference values and the output setpoint by 0.85 also provides some non-linear feedforward control.

Figure 9 shows that the concentrations of the product and the purge stream could be held almost constant over the entire time, only slightly increasing the offset indicated by the dashed lines for the setpoints. The reactor pressure shows a peak, then drops sharply and subsequently increases to the new steady-state value. Other pressures in the process follow this trend.

Figure 10 shows that the stripper bottom flow rate (production rate) drops almost immediately to its new setpoint indicated by the dashed line. The production rate drops first slightly below the setpoint and then reaches the new setpoint after only two hours of simulation time. The optimization results for the control variables are all very similar; see also Figure 10 for the control profiles of Stream 1 – 3 and the Purge Stream. The flow rates drop sharply and subsequently increase to about 85 % of their base case values. The entire TE process could be safely shifted to this new setpoint in only about 10 hours of simulated process time.

IPOPT could solve all 1000 optimization problems successfully to the desired tolerance of  $10^{-5}$ . The CPU times remain nearly constant over the entire time horizon at around 20 seconds with a minimum of 12.8 and a maximum of 24.1 seconds. The number of iterations range between 11 and 18. After switching to the new setpoint, the objective value increases to its new steady state value. The maximum constraint violation and the maximum KKT error are in the magnitude of  $10^{-6}$ .

### 4.3 Reactor Operating Pressure Change Study

The purpose of this study is to make a step change so that the operating pressure decreases from 2805 to 2745 kPa. The same approach is used as in the previous study: after 300 optimizations with the modified base case objective, OCC switches automatically to a new objective formulation. In this new objective, the pressure term in equation (14) is added,

$$(2.745 - pr\_MPa)^2 / 2.745 \quad (15)$$

while the rest of the optimization problem remains unchanged. IPOPT finds optimal solutions for all 1000 optimizations. Calculation times are in the range of 13.619 – 21.561 CPU seconds while 11 – 13 iterations were needed. The objective values show step increases and subsequent decreases to a new steady state value, which is slightly higher than the base case value. The maximal KKT error and the maximal constraint violations are in the range of  $10^{-5}$  to  $10^{-6}$ .

The goal of lowering the reactor pressure was clearly achieved as Figure 11 demonstrates. However, because integral control action was not implemented, the reactor pressure does not reach the desired setpoint of 2745 kPa. Instead, it stays slightly higher at around 2770 kPa. However, the desired setpoint change (decrease of 60 kPa) could be achieved, taking into account that the base case pressure was already higher than 2805 kPa. The optimizer could achieve this pressure reduction in only around 3 hours of simulated process time. The product concentrations and the purge stream concentration remain almost constant, as does the production rate itself, see Figure 11 and 12. The new operating setpoint therefore produces the same product amount with the same concentrations at a lower reactor pressure. All control profiles of the input and output streams return to their base case values showing only a slight effect at the time of the step change. These results mean that the TE process could be operated at a lower reactor level with the same input and output streams as in the base case.

How is this result achieved? At first guess, one would assume that the pressure is lowered by a decrease of reactor temperature and an increase in the cooling water flow. However, as Figure 13 shows, the reactor temperature remains almost constant with a change of only 0.05 K and the cooling water flow even drops first and then returns to its base case value.

The reaction rates  $R_1$ ,  $R_2$ , and  $R_3$  of equations (9)-(12)<sup>2</sup>, as shown in Figure 14, remain constant due to a decrease of the liquid level in the reactor, which results in more vapor volume available for the reactions to take place. Lowering the liquid level is achieved by lowering the reactor feed stream 6 (see Figure 15) and by the short drop in the reactor cooling water flow (see Figure 14). The exothermic heat by reactions  $R_1$ - $R_3$  evaporates the liquid until the new steady-state value is reached. The NLP solver enforces directly the constraints on the reactor liquid level as given in Table 3 through lower and upper limits on the particular variable. These variable bounds are set to reasonably high limits to prevent the tank from running dry. However, additional control measures would be required to prevent the reactor from running dry in the case of large disturbances.

The flow rates in the loop of the vapor streams (reactor feed stream 6, reactor product stream 7 and recycle stream 8) are slightly lower than in the base case, as seen in Figure 15. Due to the

---

<sup>2</sup> Reactions (11) and (12) have been combined and R3 represents the sum of these two reactions.

same total production rate, higher concentrations of the (condensable) components  $G$ ,  $H$ ,  $F$  are found in stream 7. As the separator cooling water flow is only slightly less than in the base case, relatively more vapor is condensed. As a result, there is an increased need to purify the separator underflow in the stripper with more stripper steam flow, see Figure 15.

This is an interesting result which could not be easily achieved by a SISO controller, which relies on the choice of one distinct manipulated variable (usually feed stream 1) to control reactor pressure (see McAvoy & Ye, 1994). The optimization results presented above affect a number of the outputs in a multivariable manner and show that low pressure operation level can be achieved with the same product flow and the same production concentration. The benefit of this solution is that the compressor work for stream 8 is lowered as the flow rate is lower and a lower pressure is needed in the mixing zone. This solution may be more favorable than the base case, especially if excess steam for the stripper is available.

#### 4.4 Change in Component B of Purge Gas

The purpose of this study is to make a step change so that the composition of component B in the gas purge changes from 13.82 to 15.82 %. Again, the same approach is used as in the previous studies: After 300 optimization using the modified base case objective formulation, OCC switches automatically to a new objective formulation with adjusted values for the concentration B in the purge stream. The new objective formulation is used for the remaining 700 optimizations.

IPOPT finds optimal solutions for all 1000 optimizations. Calculation times are in the range of only 11.5 – 39.8 CPU seconds while 11 – 33 iterations were needed. After this step change, the iteration count and CPU seconds needed for convergence increased much more than for the other two setpoint change studies. The objective values show step increases and subsequently decreases to a new steady state value, which is slightly higher than the base case value. The maximal KKT error and the maximal constraint violations are in the range of  $10^{-4}$  to  $10^{-6}$ .

The strategy found by the optimizer is shown in Figure 16. The purge stream valve closes completely for a short amount of time and then increases subsequently to its new steady-state value. The inert gas  $B$  accumulates in this time to a new concentration of about 15.15 %. Again, integral action was not implemented here to eliminate offset. Nevertheless, although the setpoint of 15.82 % is not reached exactly, the optimization problem can be considered as solved.

## 5 Conclusions

This study presents the software platform OptControlCentre (OCC) for the implementation of the on-line real-time dynamic optimization. Differential Algebraic Equation (DAE) systems are discretized automatically into finite time elements to form nonlinear programming (NLP) problems, which are subsequently converted into FORTRAN codes using the Maple-based code generator OCOMA. In order to show the performance of this approach, the on-line dynamic optimization of the Tennessee Eastman (TE) Challenge Process is considered. The DAE model of this process has almost 200 DAEs and it is discretized to an NLP of nearly 11000 system variables, 50000 nonzero entries in the Jacobian and 15000 nonzero entries in the Hessian.

Two SQP-type codes, SNOPT and IPOPT, were considered in this study. SNOPT solves a reduced QP subproblem using an active set strategy. SNOPT applies a line search algorithm based on an augmented Lagrangian function and uses a limited memory BFGS update to approximate second order information as well as an efficient reduced space decomposition to improve the solution of the QP subproblem. In contrast, IPOPT proceeds by solving a barrier NLP problem to relaxed tolerances, for decreasing values of the barrier parameter  $\mu$ . To solve the barrier problem, IPOPT takes Newton steps in the full space, using a sparse indefinite symmetric linear solver. Exact second derivatives (provided by OCOMA) are calculated for the Hessian. To ensure global convergence a filter line search method is used and second order corrections are also used to maintain a fast convergence rate close to the solution. Finally, because dynamic optimization problems for NMPC are solved repeatedly, a warm start feature has also been implemented into IPOPT. This feature allows the specification of  $\mu$  at a low restart value (e.g.  $\mu = 10^{-4}$ ) and it allows to reuse the values of Lagrange multipliers from a previous optimization.

The study has been made using four scenarios of the TE process: the base-case and a set of three setpoint changes for case studies suggested by Downs and Vogel. Each setpoint change study was simulated and optimized over 1000 cycles with a sampling time of 100 seconds (over 27 hours in total) to see the full results on the process. The performance of SNOPT was insufficient for the base-case study; most optimizations took well over 1000 seconds and many did not converge within the limiting 100 major iterations.

In contrast, the interior point solver IPOPT is well suited for large-scale dynamic optimization and NMPC formulations. Calculation times of IPOPT for the setpoint change studies are in the range of only 11.5 – 39.8 CPU seconds while 11 – 33 iterations were needed. The majority of optimization cases required only 20 CPU seconds and 12 iterations. The maximal KKT error and the maximal constraint violations are in the range of  $10^{-4}$  to  $10^{-6}$ . The online optimization was executed on a Pentium III, 1GHz with 512 MB RAM. Solutions for the setpoint change studies show performance changes that could not be easily achieved by a conventional SISO control scheme. For example, the solutions from IPOPT suggest an internal shift in the TE process so that the same production rate with the same product concentrations and input feed streams could be achieved at a lower reactor operating pressure. The setpoint change studies show some zero-state offset. To deal with steady-state offset, one would normally add integrating action. This leads to additional linear equations that would slightly affect the linear algebra but would not affect the overall performance of the optimization code.

Future work will include extending the above approach to PDE systems and the development of partial solution strategies for even faster CPU times. We will further incorporate the dynamic process model of the actual plant model and consider dynamic system identification prior to the NMPC optimization.

## 6 Acknowledgements

This research project was funded by a personal Ph.D. scholarship for Tobias Jockenhövel from the Ernst von Siemens-Foundation. The authors gratefully acknowledge the work of Rolf Feller in constructing a version of the Tennessee Eastman simulation model.

## 7 Nomenclature

The reader is referred to (Jockenhövel, 2002) for the model file which contains the values of all model constants.

$C_{p,cw}$	$\text{kJ kg}^{-1} \text{K}^{-1}$	Specific heat capacity cooling water
$C_{p,liq,i}$	$\text{kJ kg}^{-1} \text{K}^{-1}$	Specific heat capacity component $j$ in liquid phase
$C_{p,vap,i}$	$\text{kJ kg}^{-1} \text{K}^{-1}$	Specific heat capacity component $j$ in vapor phase
$F_j$	$\text{kg h}^{-1}$	Flow rate of stream $j$
$H_i$	$\text{kJ}$	Enthalpy of component $i$
$H_o$	$\text{kJ}$	Reference enthalpy
$m_{cw,r}$	$\text{kg h}^{-1}$	Cooling water flow rate reactor
$m_{cw,s}$	$\text{kg h}^{-1}$	Cooling water flow rate separator
$N_{i,m}$	$\text{Kmol}$	Moles of component $i$ in mixer
$N_{i,r}$	$\text{Kmol}$	Moles of component $i$ in reactor
$N_{i,s}$	$\text{Kmol}$	Moles of component $i$ in separator
$N_{i,str}$	$\text{Kmol}$	Moles of component $i$ in stripper
$p_{i,r}$	$\text{kPa}$	Partial pressure of component $i$ in reactor
$p_i^{sat}$	$\text{kPa}$	Saturation pressure of component $i$
$p_m$	$\text{kPa}$	Pressure in mixer
$p_r$	$\text{kPa}$	Pressure in reactor
$p_s$	$\text{kPa}$	Pressure in separator
$Q_r$	$\text{kW}$	Energy removed from the reactor
$Q_s$	$\text{kW}$	Energy removed from the separator
$Q_{str}$	$\text{kW}$	Energy added to the stripper
$R$	$\text{kmol h}^{-1}$	Reaction conversion rate
$R_j$	$\text{kmol h}^{-1}$	Reaction conversion component $j$
$T_{cw,r,in}$	$\text{K}$	Cooling water inlet temperature reactor
$T_{cw,r,out}$	$\text{K}$	Cooling water outlet temperature reactor
$T_{cw,s,in}$	$\text{K}$	Cooling water inlet temperature separator

$T_{CW,s,out}$	K	Cooling water outlet temperature separator
$T_m$	K	Temperature mixer
$T_r$	K	Temperature reactor
$T_s$	K	Temperature separator
$T_{str}$	K	Temperature stripper
$t$	S	Time
$V_{L,r}$	$m^3$	Volume of liquid phase in reactor
$V_m$	$m^3$	Volume mixer
$V_s$	$m^3$	Volume separator
$V_{str}$	$m^3$	Volume stripper
$V_{V,r}$	$m^3$	Volume vapor phase reactor
$V_{l,str}$	$m^3$	Volume liquid phase stripper
$\nu_{i,j}$	–	Stoichiometric reaction coefficient
$x_{i,r}$	$mol\ mol^{-1}$	Liquid concentration of component $i$ in reactor
$y_{i,j}$	$mol\ mol^{-1}$	Vapor concentration of component $i$ in stream $j$
$\Delta H_{Rj}$	$kJ\ kmol^{-1}$	Exothermic heat
$\mathbf{a}_j$	$kmol\ h^{-1}\ m^{-3}$	Reaction constant
UA	$kW\ K^{-1}$	Specific heat transfer rate
$\mathbf{g}_i$	–	Activity coefficient
$\mathbf{r}_{str}$	$kg\ m^{-3}$	Density in stripper
$\Phi_i$	$mol\ mol^{-1}$	Split fraction of component $i$

## 8 References

Albuquerque, J.; Gopal V., Staus G., Biegler L. T. & Ydstie, B. E. (1999). Interior point SQP strategies for large-scale, structured process optimization problems. *Comp. Chem. Eng.*, 23(4-5), 543-554.

Aspen Technology (2002). Aspen Custom Modeler User's Guide. <http://www.aspentech.com>

Bausa, J. (2000). Dynamische Optimierung energie- und verfahrenstechnischer Prozesse. *Ph.D. Thesis*. Technical University of Berlin, Germany.

Betts, J. T. (2001). Practical Methods for Optimal Control Using Nonlinear Programming. Advances in Design and Control 3, SIAM, Philadelphia, U.S.A.

Biegler, L. T. (2000). Efficient Solution of Dynamic Optimization and NMPC Problems. 219-245, *Nonlinear Model Predictive Control*, Allgoewer, F., & Zheng, A. (eds.). Basel: Birkhaeuser.

Biegler, L.T., A. M. Cervantes and A. Wächter (2002). Advances in Simultaneous Strategies for Dynamic Process Optimization, *Chem. Eng. Science*. **57**(4), 575-593.

Binder, T., Blank, L., Bock, H. G., Bulitsch, R., Dahmen, W., Diehl, M., Kronseder, T., Marquardt, W., Schloeder J., von Stryk O. (2001). Introduction to Model Based Optimization of Chemical Processes on Moving Horizons. In Groetschel, M., Krumke, S. O., & Rambau, J. (eds.). *Online Optimization of Large Scale Systems*. Berlin: Springer.

Cervantes, A. M., Wächter A., Tütüncü R. H., & Biegler, L. T. (2000). A reduced space interior point strategy for optimization of differential algebraic systems. *Comp. Chem. Eng.*, 24(1), 39-51, 2000.

Diehl, M., Findeisen, R., Nagy, Z., Bock, H.G., Schlöder, J.P., & Allgöwer, F. (2002). Real-time optimization and nonlinear model predictive control of processes governed by differential-algebraic equations. To appear in J. Proc. Contr.

Downs, J. J., & Vogel, E. F. (1993). A plant-wide industrial process control problem. *Comp. Chem. Eng.*, 17 (3), 245-255.

Duvall, P. M., & Riggs, J. B. (2000). Online optimization of the Tennessee Eastman challenge problem, *J. Process Control*, 10(1), 19-33.

Feller, R. (1997). Nonlinear Model Predictive Control of the Tennessee Eastman Challenge Process. *Student Senior Project*. Carnegie Mellon University, U.S.A.

Gill, P. E., Murray W., & Saunders, M. A. (1998). User's guide for SNOPT: A FORTRAN package for large-scale nonlinear programming. Technical report, Department of Mathematics, University of California, San Diego, USA.

Groetschel, M., Krumke, S. O., & Rambau J. (eds.) (2001). *Online Optimization of Large Scale Systems*. Berlin: Springer.

Jockenhövel, T. (2002). OptControlCentre, Software package for dynamic optimization. <http://www.OptControlCentre.com/>

Jockenhövel, T. (2003). *Ph.D. Thesis* (in preparation). Technical University of Berlin, Germany.

Larsson T., T. & Skogestad S. (2000). Plantwide control: A review and a new design procedure" *Modeling, Identification and Control*, 21, 209-240.

Larsson, T., Hestetun K., Hovland E. & Skogestad S. (2001). Self-Optimizing Control of a Large-Scale Plant: The Tennessee Eastman Process, *Ind. Eng. Chem. Res.*, 40 (22), 4889 -4901.

Leineweber D. B., Bock, H.G., Schlöder, J.P., Gallitzendörfer, J.V., Schäfer, A., & Jansohn, P. (1997). A Boundary Value Problem Approach to the Optimization of Chemical Processes Described by DAE Models. Submitted to *Comp. Chem. Eng.*

Mayne, D. Q., Rawlings, J. B., Rao, C. V., & Scokaert, P. (2000). Constrained model predictive control: Stability and optimality. *Automatica*, 36(6), 789-814.

McAvoy, T. J., & Ye, N. (1994). Base Control for the Tennessee Eastman Problem. *Chem. Eng.*, 18 (5), 383-413.

Process Systems Enterprises Limited (2002). gProms User's Guide. <http://www.psenterprise.com/>

Ricker, N. L. (1995). Optimal steady state operation of the Tennessee Eastman challenge process. *Comp. Chem. Eng*, 19 (9), 949-959.

Ricker, N. L., & Lee, J. H. (1995a). Nonlinear model predictive control of the Tennessee Eastman challenge process. *Comp. Chem. Eng*, 19 (9), 961-981.

Ricker, N. L., & Lee, J. H. (1995b). Nonlinear modeling and state estimation for the Tennessee Eastman challenge process. *Comp. Chem. Eng*, 19 (9), 983-1005.

Schittkowski, K. (1987). *More test examples for nonlinear programming codes*. Lecture notes in economics and mathematical systems # 282. Berlin: Springer.

Vassiliadis, V. R., Sargent, W. H., & Pantelides, C. (1994). Solution of a class of multistage dynamic optimization problems. 2. Problems with Path Constraints. *Ind. Eng. Chem. Research.*, **33**(9), 2123-2133.

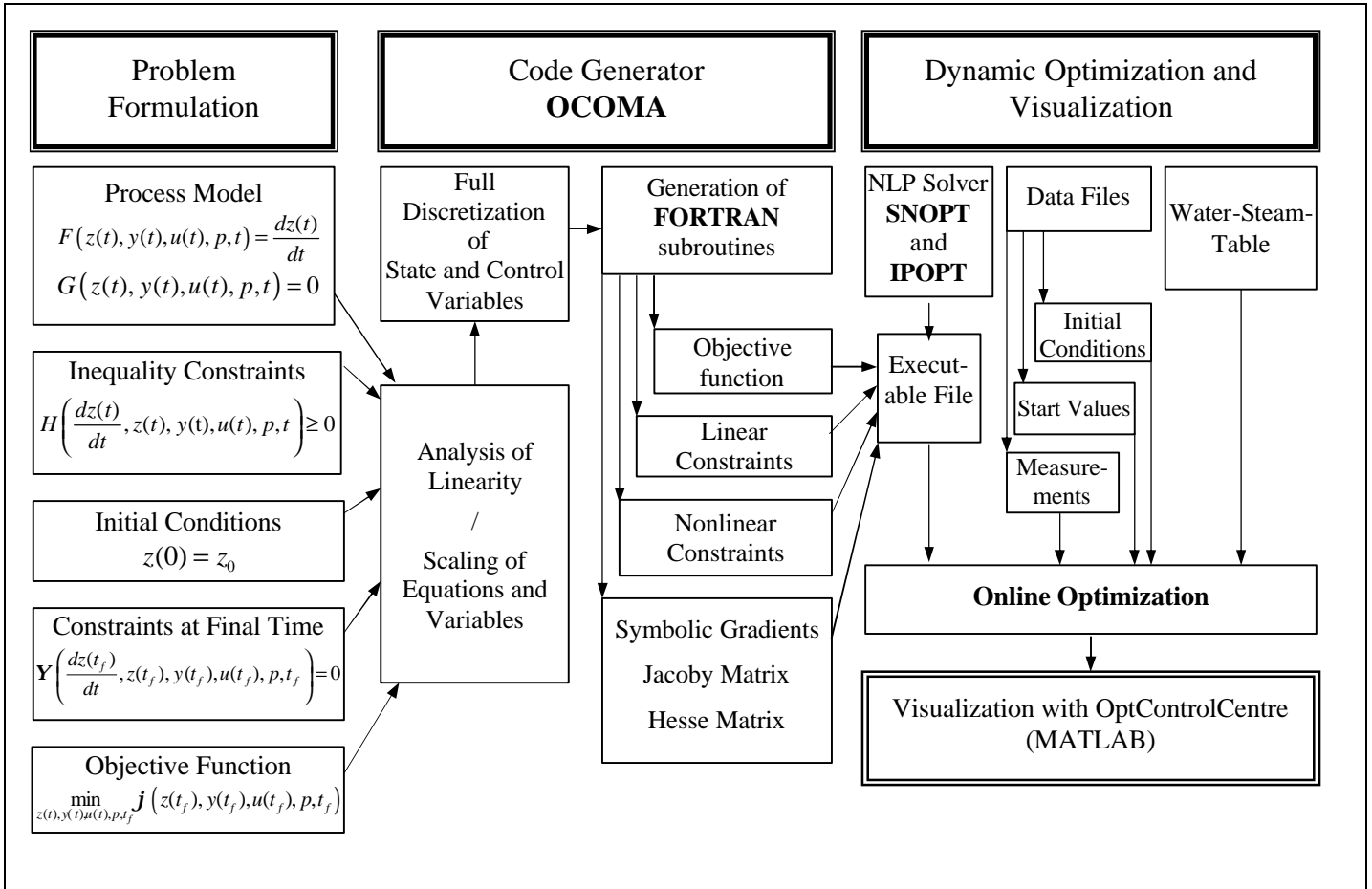
Wächter, A. (2002). An Interior Point Algorithm for Large-Scale Nonlinear Optimization with Applications in Process Engineering. *Ph.D. Thesis*, Carnegie Mellon University, U.S.A. (IPOPT and documentation are available from <http://www.coin-or.org>)

Yan, M., & Ricker, N.L., On-line optimization of the Tennessee Eastman challenge process., In: Proceedings of the 1997 American Control Conference (Cat. No.97CH36041); Evanston, IL, USA : American Autom. Control Council, 1997, 6 vol. p. (2960-5 vol.5) Part No: vol.5

Zheng, A., & Zhang W. (2001). Computationally efficient nonlinear model predictive control algorithm for control of constrained nonlinear systems. xiv+261, 173-87. In: Kouvaritakis, B. Cannon, M.; Nonlinear predictive control theory and practice; London, UK: IEE.

## 9 List of Figures

Figure 1 OCOMA for Online Optimization .....	20
Figure 2 The Tennessee Eastman Process.....	21
Figure 3 Mixing zone process model .....	22
Figure 4 Reactor process model.....	23
Figure 5 Separator and compressor models.....	24
Figure 6 Stripper process model. ....	25
Figure 7 Formulation of the objective (OCOMA code).....	26
Figure 8 Objective function and reactor pressure in the modified base case. ....	26
Figure 9 Product concentration (with setpoints) and reactor pressure in setpoint change study 1.....	27
Figure 10 Various control profiles for setpoint change study 1 .....	28
Figure 11 Reactor pressure, product concentration and production rate in setpoint change study 2 .....	29
Figure 12 Input and output stream trajectories of setpoint change study 2.....	30
Figure 13 Reactor temperature and cooling water flow, setpoint change study 2.....	31
Figure 14 Reaction rates and reactor liquid level, setpoint change study 2 .....	31
Figure 15 Internal process streams, separator cooling water flow and stripper steam flow in setpoint change study 2.....	32
Figure 16 Purge stream flow rate and concentration B, setpoint change study 3.....	33



**Figure 1** OCOMA for Online Optimization

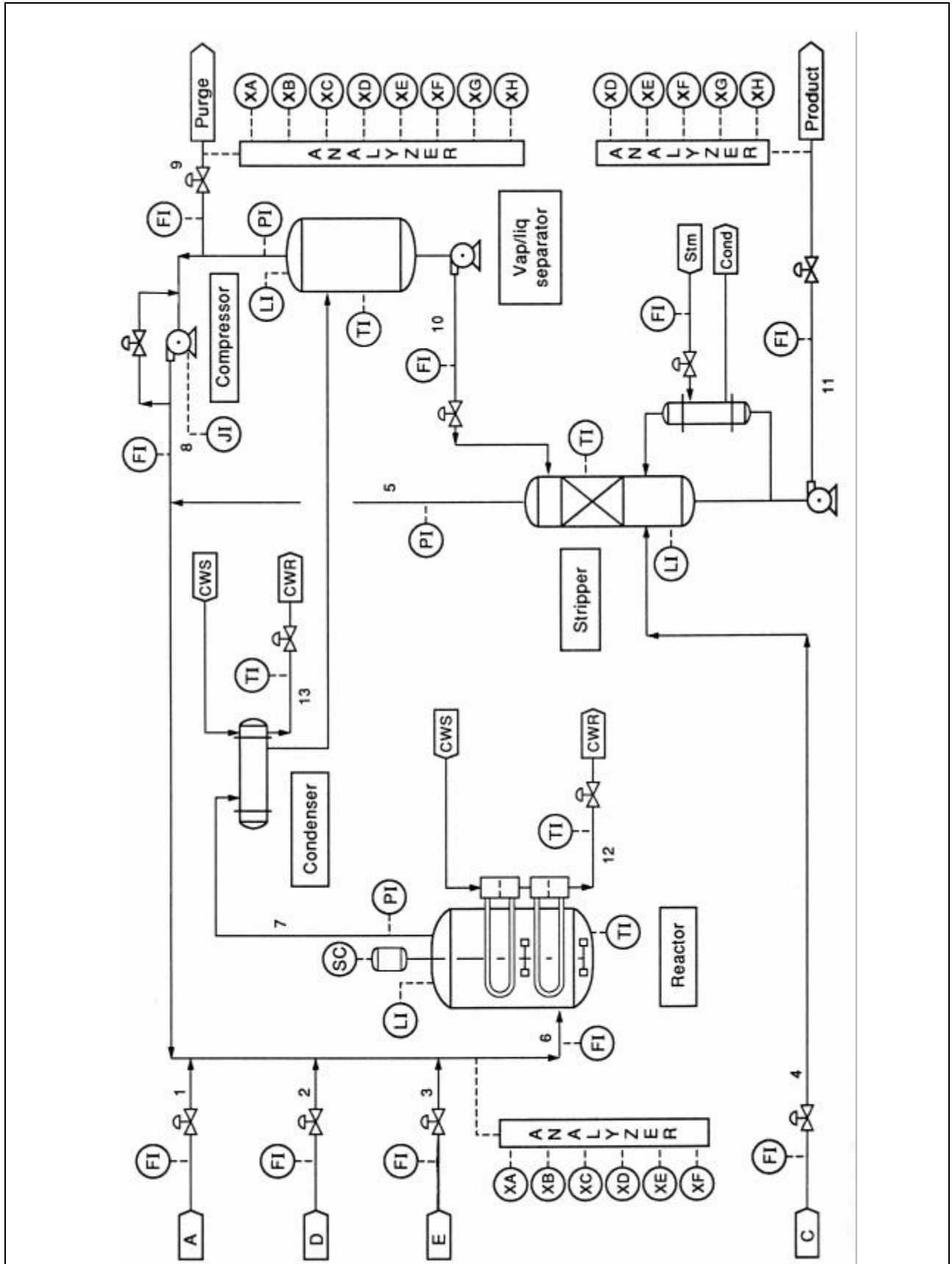


Figure 2 The Tennessee Eastman Process

Molar Balances for components A - H

$$\frac{dN_{i,m}}{dt} = y_{i,1}F_1 + y_{i,2}F_2 + y_{i,3}F_3 + y_{i,5}F_5 + y_{i,8}F_8 - y_{i,6}F_6$$

Energy balance for the mixing zone

$$\left( \sum_{i=A}^H N_{i,m} c_{p,vap,i} \right) \frac{dT_m}{dt} = \sum_{j=1,2,3,5,8} F_j \left( \sum_{i=A}^H y_{i,j} c_{p,vap,i} \right) (T_j - T_m)$$

Pressure and concentrations

$$y_{i,6} = \frac{N_{i,m}}{\sum_{j=A}^H N_{j,m}}; \quad p_m = \sum_{i=A}^H N_{i,m} \frac{RT_m}{V_m}$$

**Figure 3** Mixing zone process model

Molar Balances for components A-H

$$\frac{dN_{i,r}}{dt} = y_{i,6}F_6 - y_{i,7}F_7 + \sum_{j=1}^3 \mathbf{u}_{ij}R_j \quad i = A, \dots, H$$

Energy balance for the reactor

$$\left( \sum_{i=A}^H N_{i,r} c_{p,i} \right) \frac{dT_r}{dt} = F_6 \left( \sum_{i=A}^H y_{i,6} c_{p,vap,i} \right) (T_6 - T_r) - \dot{Q}_r - \sum_{j=1}^3 \Delta H_{Rj} R_j$$

$$\Delta H_{Rj} = \sum_{i=A}^H H_i \mathbf{u}_{ij} + H_o F_j, \text{ with } H_i = c_{p,i} (T_r - T^*)$$

Reaction kinetics

$$R_1 = \mathbf{a}_1 V_{V,r} \exp \left[ 44.06 - \frac{42600}{RT_r} \right] p_{A,r}^{1.08} p_{C,r}^{0.311} p_{D,r}^{0.874}$$

$$R_2 = \mathbf{a}_2 V_{V,r} \exp \left[ 10.27 - \frac{19500}{RT_r} \right] p_{A,r}^{1.15} p_{C,r}^{0.370} p_{E,r}^{1.00}$$

$$R_3 = \mathbf{a}_3 V_{V,r} \exp \left[ 59.50 - \frac{59500}{RT_r} \right] p_{A,r} (0.77 p_{D,r} + p_{E,r})$$

Reactor input stream F6 and reactor output stream F7

$$F_6 = 0.8334 \frac{\text{kmol}}{\text{s}\sqrt{\text{MPa}}} \sqrt{p_m - p_r}$$

$$F_7 = 1.5355 \frac{\text{kmol}}{\text{s}\sqrt{\text{MPa}}} \sqrt{p_r - p_s}$$

The constants values are chosen to match the base case. Note that the cooling water flux is a control variable directly. Its dependence on a valve position is neglected. Note that for components D-H the numbers of moles  $N_{i,r}$  refer to the number of moles in the liquid phase only, due to the assumption that a buildup of these components in the vapor phase can be neglected.

Heat exchange with cooling water

$$\dot{Q}_r = m_{CW,r} c_{p,CW} (T_{CW,r,out} - T_{CW,r,in})$$

$$\dot{Q}_r = UA_r \left( \frac{\Delta T_1 - \Delta T_2}{\ln(\Delta T_1 / \Delta T_2)} \right)$$

$$\Delta T_1 = T_r - T_{CW,r,in}; \quad \Delta T_2 = T_r - T_{CW,r,out}$$

Vapor-liquid equilibrium

$$p_{i,r} = \mathbf{g}_{i,r} x_{i,r} p_{i,r}^{sat}(T_r) \quad i = D, \dots, H$$

$$p_{i,r} = \frac{N_{i,r,vap} RT_r}{V_{V,r}} \quad i = A, B, C$$

$$p_{i,r}^{sat}(T) = 10^{-3} \exp \left( A_i + \frac{B_i}{C_i + T_r - T^*} \right) \quad i = D, \dots, H$$

$$p_r = \sum_{i=A}^H p_{i,r}$$

$$y_{i,7} = \frac{p_{i,r}}{p_r} \quad i = A, \dots, H$$

$$x_{i,r} = \frac{N_{i,r}}{\sum_{i=D}^H N_{i,r}} \quad i = D, \dots, H$$

$$V_{L,r} = \frac{\sum_{i=D}^H N_{i,r,liq}}{\mathbf{r}_{liq,r}}, \text{ with } V_{V,r} = V_r - V_{L,r}$$

**Figure 4** Reactor process model.

Molar Balances for components A-H

$$\frac{dN_{i,s}}{dt} = y_{i,7} F_7 - y_{i,8} (F_8 + F_9) - x_{i,s} F_{10} \quad i = A, \dots, H$$

Energy balance for the separator

$$\left( \sum_{i=A}^H N_{i,s} c_{p,i} \right) \frac{dT_s}{dt} = F_7 \left( \sum_{i=A}^H y_{i,7} c_{p,vap,i} \right) (T_r - T_s) + HoV_s - \dot{Q}_s$$

$$HoV_s = \sum_{i=D}^H x_{i,10} F_{10} \cdot H_{vap,i}$$

Vapor-liquid equilibrium

$$p_{i,s} = \frac{N_{i,s,vap} RT_s}{V_{V,s}} \quad i = A, B, C; \quad p_s = \sum_{i=A}^H p_{i,s}; \quad p_{i,s} = g_{i,s} x_{i,10} p_{i,s}^{sat}(T_s) \quad i = D, \dots, H$$

$$y_{i,8} = y_{i,9} = \frac{p_{i,s}}{p_s}; \quad x_{i,10} = 0 \quad i = A, B, C; \quad x_{i,10} = \frac{N_{i,s}}{\sum_{i=D}^H N_{i,s}} \quad i = D, \dots, H; \quad V_{L,s} = \frac{\sum_{i=D}^H N_{i,s,liq}}{\mathbf{r}_{liq,s}}; \quad V_{V,s} = V_s - V_{L,s}$$

Compressor model (Temperature change due to compressor work)

$$T_8 = T_s \left( \frac{p_m}{p_s} \right)^{\frac{1-k}{k}}$$

Heat exchange with cooling water

$$\dot{Q}_s = m_{CW,s} c_{p,CW} (T_{CW,s,out} - T_{CW,s,in})$$

$$\dot{Q}_s = UA_s \left( \frac{\Delta T_1 - \Delta T_2}{\ln(\Delta T_1 / \Delta T_2)} \right)$$

$$\Delta T_1 = T_s - T_{CW,s,in};$$

$$\Delta T_2 = T_s - T_{CW,s,out}$$

**Figure 5** Separator and compressor models.

Molar Balances for components  $G$  and  $H$

$$\frac{dN_{i,stri}}{dt} = (1 - \Phi_i) (x_{i,10} F_{10} + y_{i,4} F_4) - x_{i,11} F_{11} \quad i = G, H$$

Energy balance for the stripper

$$\left( \sum_{i=G}^H N_{i,stri} c_{p,liq,i} \right) \frac{dT_{stri}}{dt} = F_{10} \sum_{i=A}^H (x_{i,10} c_{p,liq,i}) (T_s - T_{stri}) + F_4 \sum_{i=A}^H (y_{i,4} c_{p,vap,i}) (T_4 - T_{stri}) - HoV_{stri} + \dot{Q}_{stri}$$

$$HoV_{stri} = \sum_{i=D}^H H_{vap,i} (y_{i,5} F_5 - y_{i,4} F_4); \quad \dot{Q}_{stri} = 2258.717 \frac{\text{kJ}}{\text{kg}} \dot{m}_{steam}$$

Vapor-liquid equilibrium

$$V_{l,stri} = \sum_{i=D}^H \frac{N_{i,stri}}{r_{stri}}; \quad \Phi_i = \sum_{j=0}^3 a_{i,j} (T_s - 273)^j \quad i = D, \dots, H; \quad \Phi_i = 1 \quad i = A, B, C,$$

$$F_5 = F_{10} + F_4 - F_{11} - \sum_{i=G}^H \frac{dN_{i,stri}}{dt}; \quad y_{i,5} = \frac{\Phi_i (y_{i,4} F_4 + x_{i,10} F_{10})}{F_5} \quad i = A, \dots, H$$

$$x_{i,11} = \frac{y_{i,4} F_4 + x_{i,10} F_{10} - y_{i,5} F_5}{F_{11}} \quad i = D, F; \quad x_{i,11} = \left( 1 - \sum_{j=D}^F x_{j,11} \right) \frac{N_{i,stri}}{\sum_{j=D}^H N_{j,stri}} \quad i = G, H$$

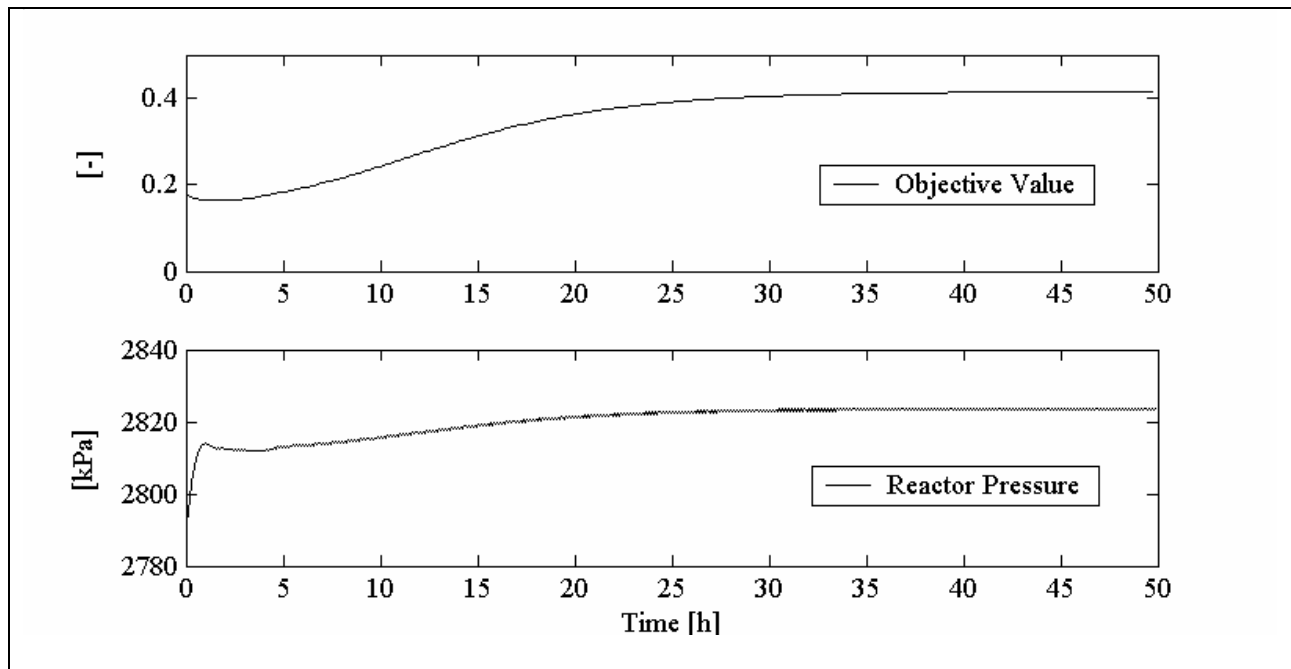
**Figure 6** Stripper process model.

```

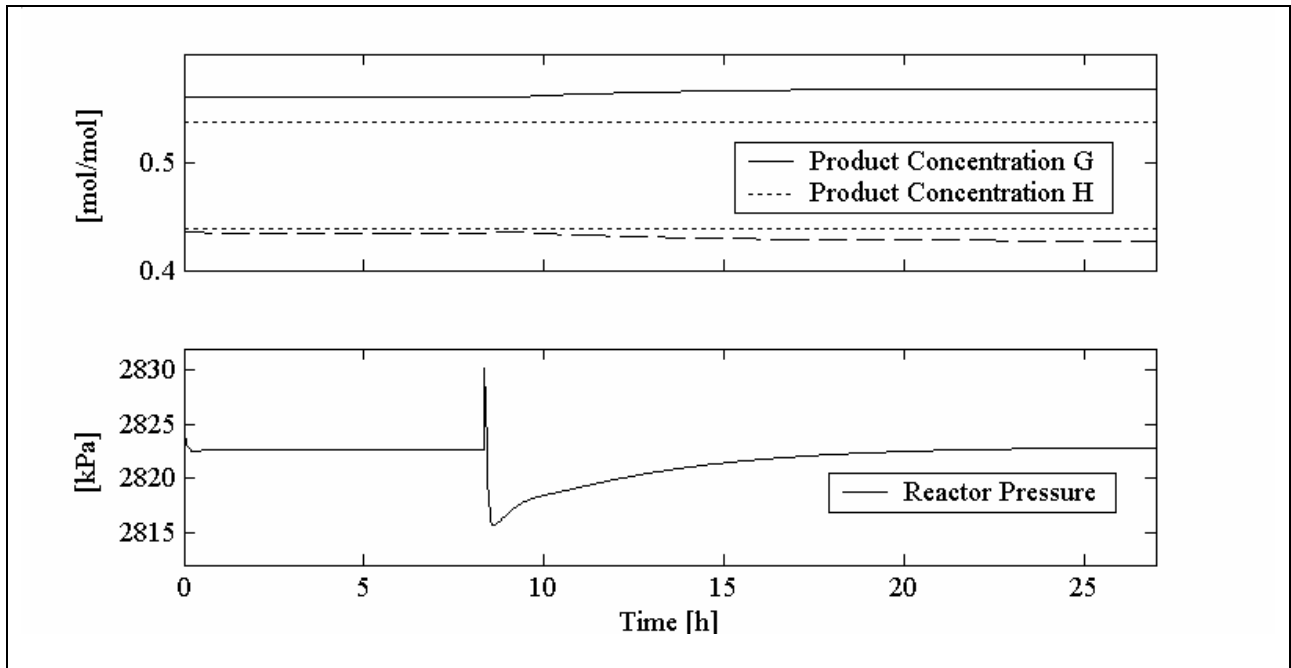
objective( ( 0.13823 - stream_8_conc_B )^2 / 0.13823 +
           ( 0.53724 - stream_11_conc_G )^2 / 0.53724 +
           ( 0.43828 - stream_11_conc_H )^2 / 0.43828 +
           0.1 * ((11.2 / 3600) - stream_1_flow )^2 / (11.2 / 3600) +
           0.1 * ((114.5 / 3600) - stream_2_flow )^2 / (114.5 / 3600) +
           0.1 * ((98 / 3600) - stream_3_flow )^2 / (98 / 3600) +
           0.1 * ((417.5 / 3600) - stream_4_flow )^2 / (417.5 / 3600) +
           0.1 * ((1201.5 / 3600) - stream_8_flow )^2 / (1201.5 / 3600) +
           0.1 * ((15.1 / 3600) - stream_9_flow )^2 / (15.1 / 3600) +
           0.1 * ((259.5 / 3600) - stream_10_flow)^2 / (259.5 / 3600) +
           0.1 * ((211.3 / 3600) - stream_11_flow)^2 / (211.3 / 3600) +
           0.1 * ( 25.93611 - reactor_cooling_water )^2 / 25.93611 +
           0.1 * ( 13.71389 - separator_cooling_water )^2 / 13.71389 +
           0.1 * ( 0.063975 - stripper_steam )^2 / 0.063975 , discrete);

```

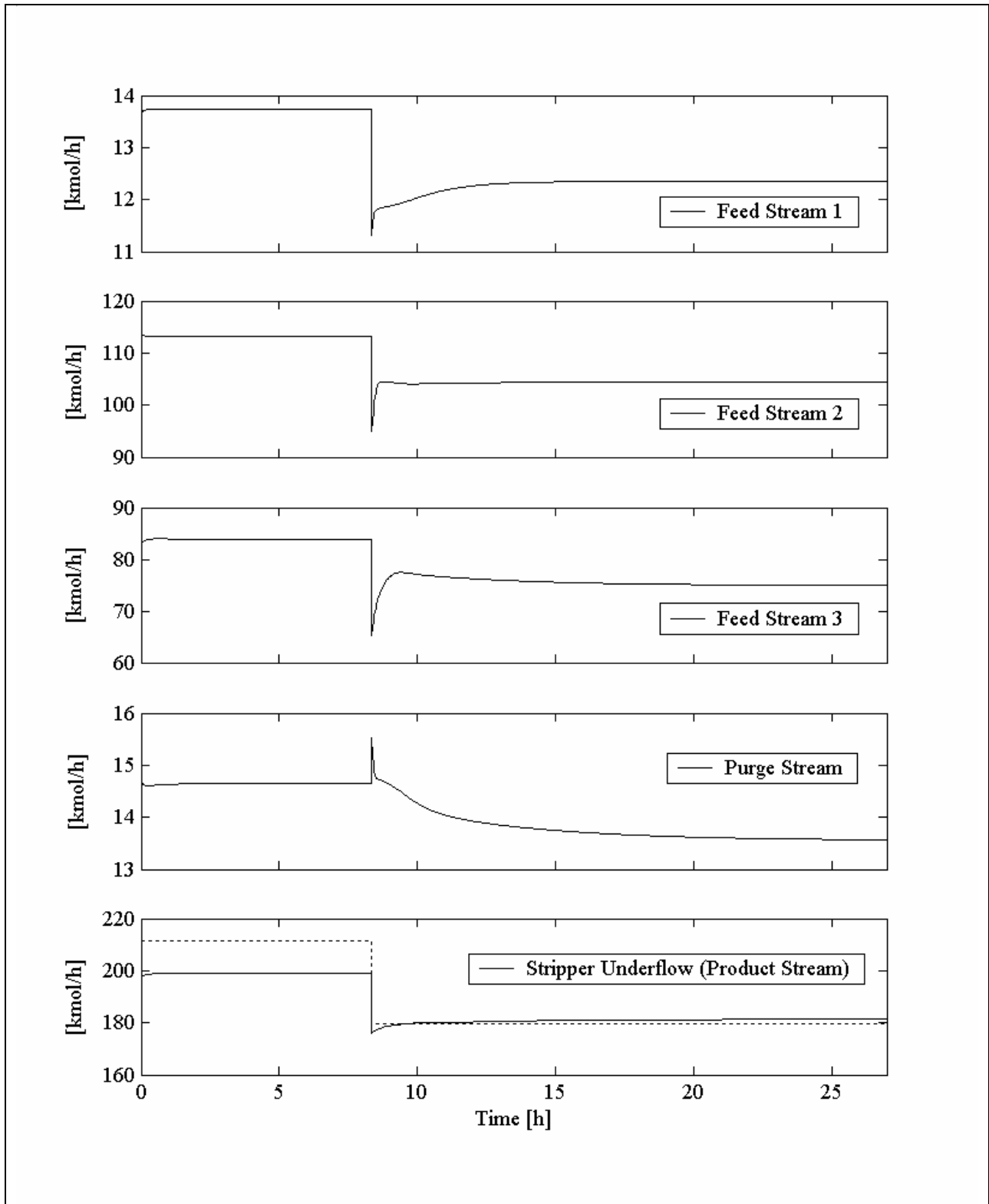
**Figure 7** Formulation of the objective (OCOMA code)



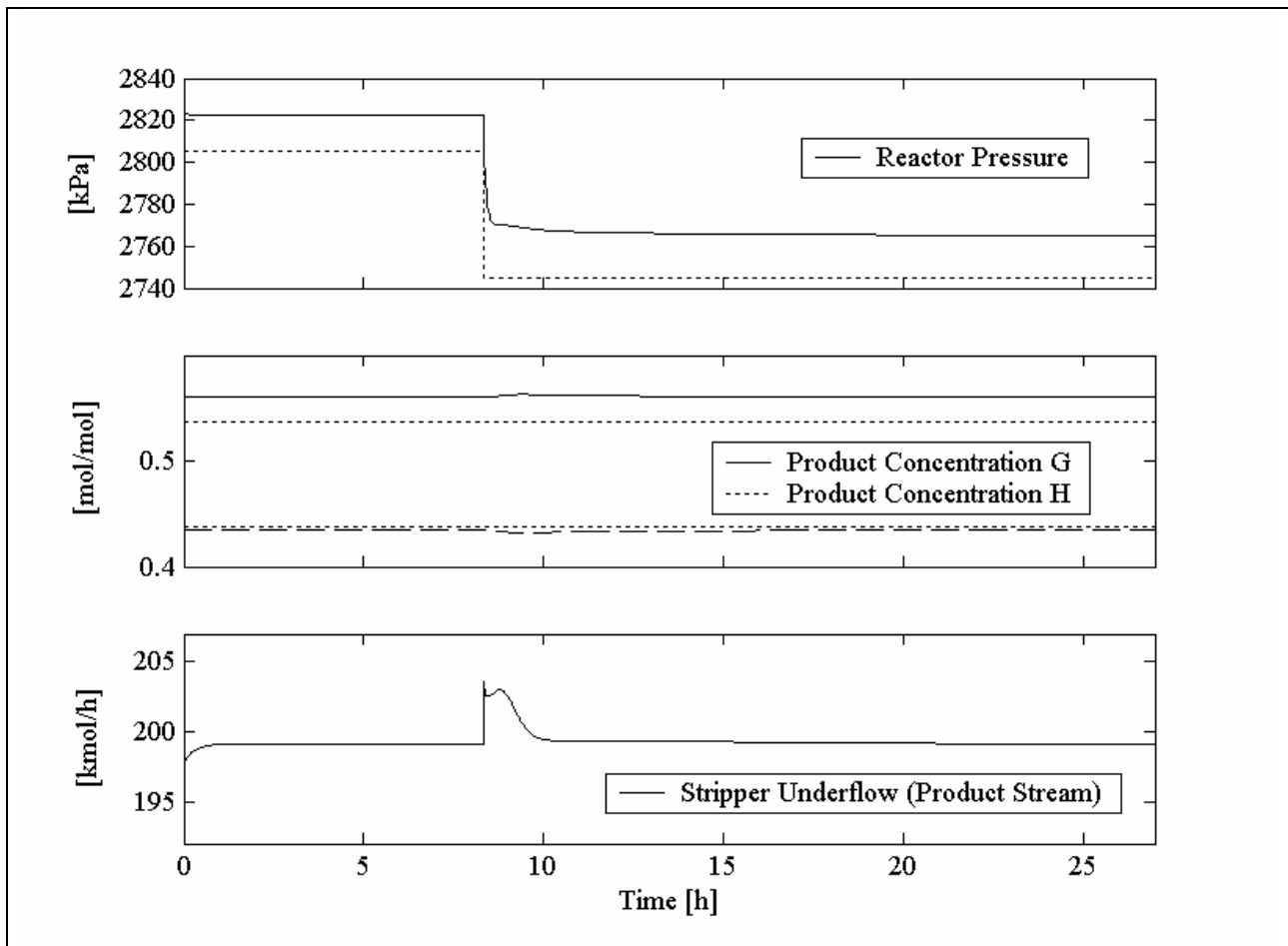
**Figure 8** Objective function and reactor pressure in the modified base case.



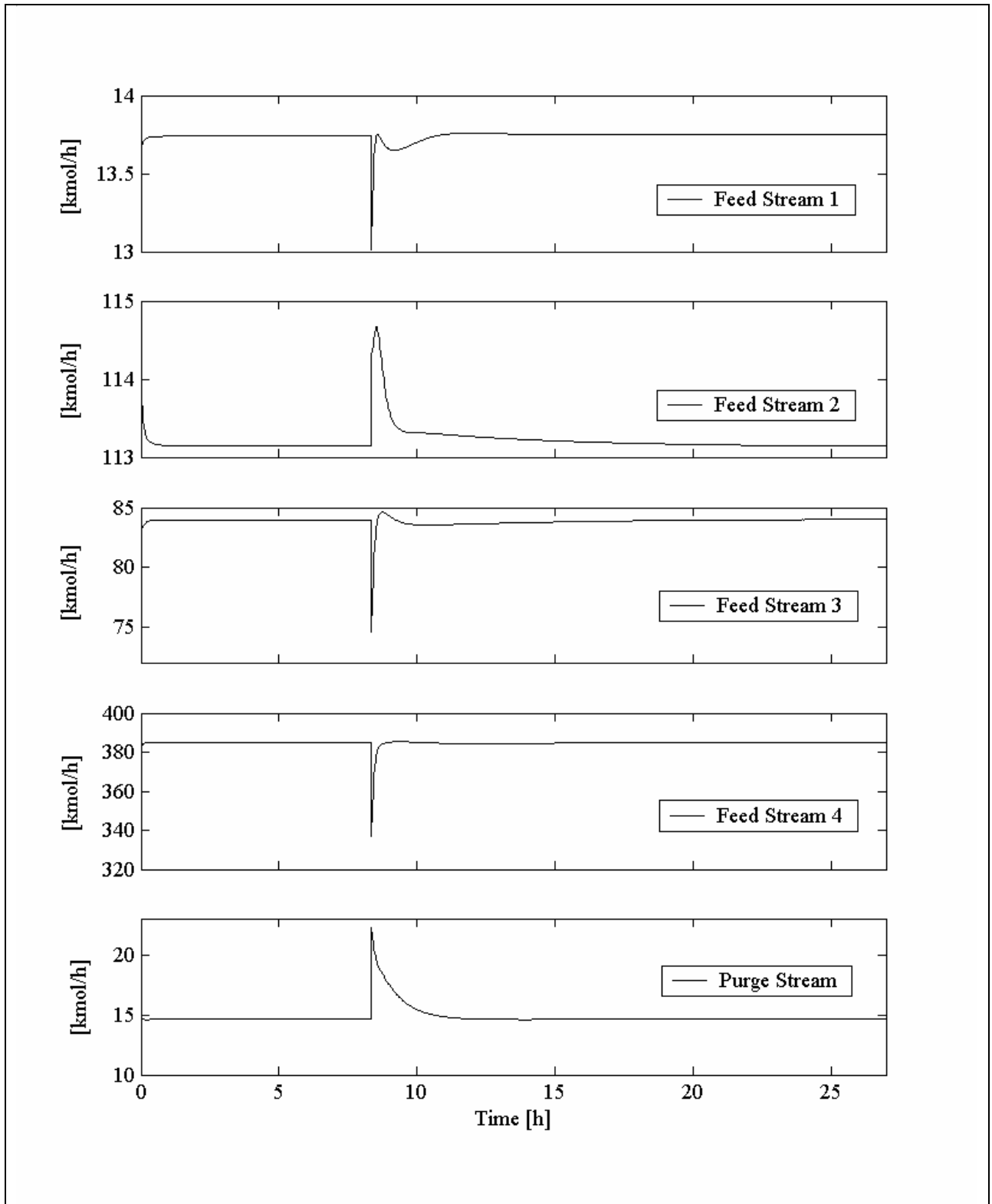
**Figure 9** Product concentration (with setpoints) and reactor pressure in setpoint change study 1.



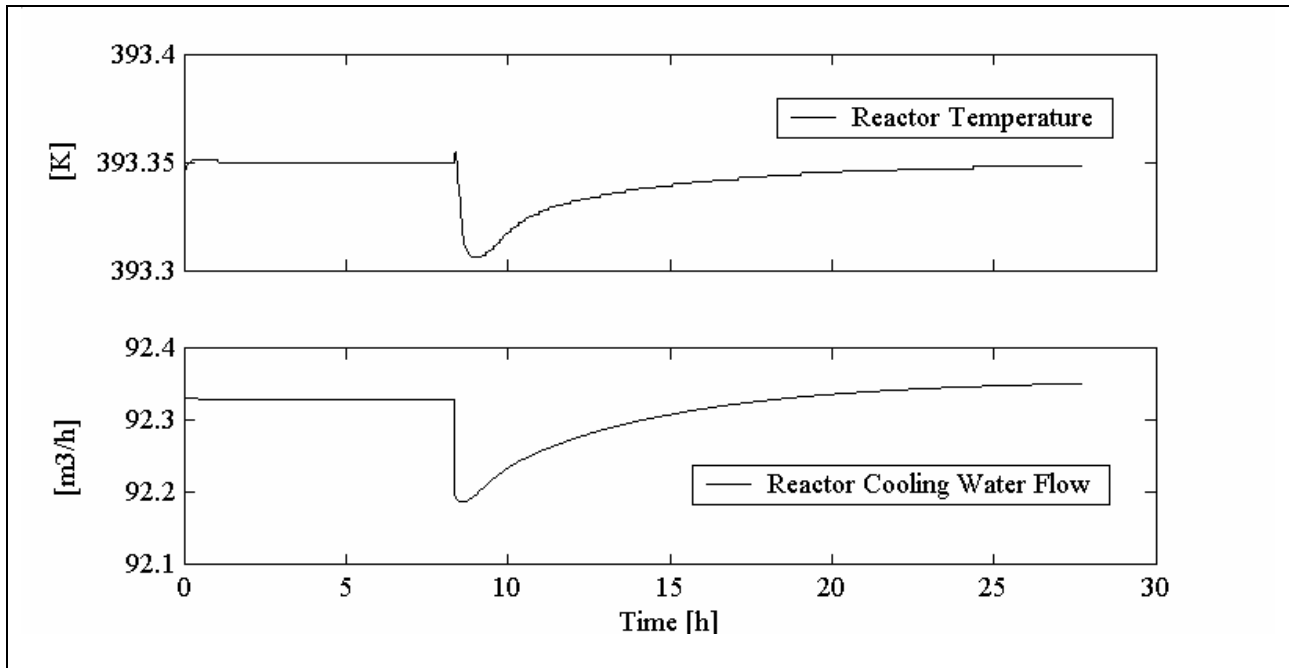
**Figure 10** Various control profiles for setpoint change study 1



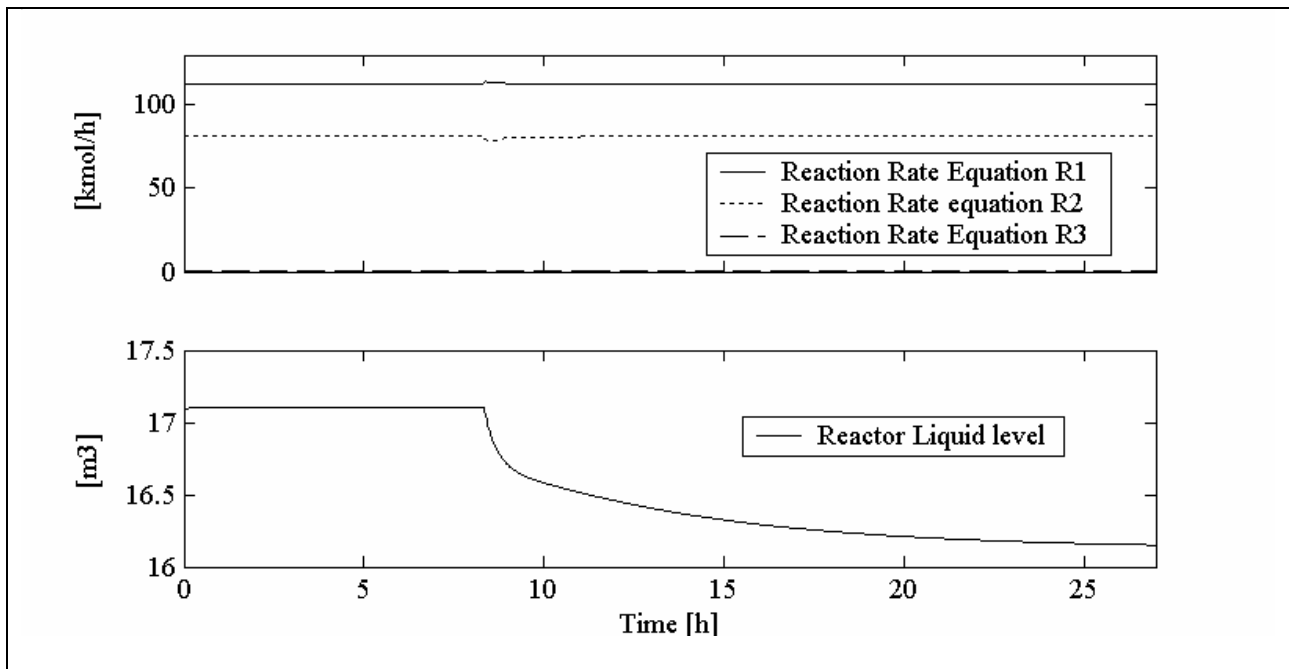
**Figure 11** Reactor pressure, product concentration and production rate in setpoint change study 2



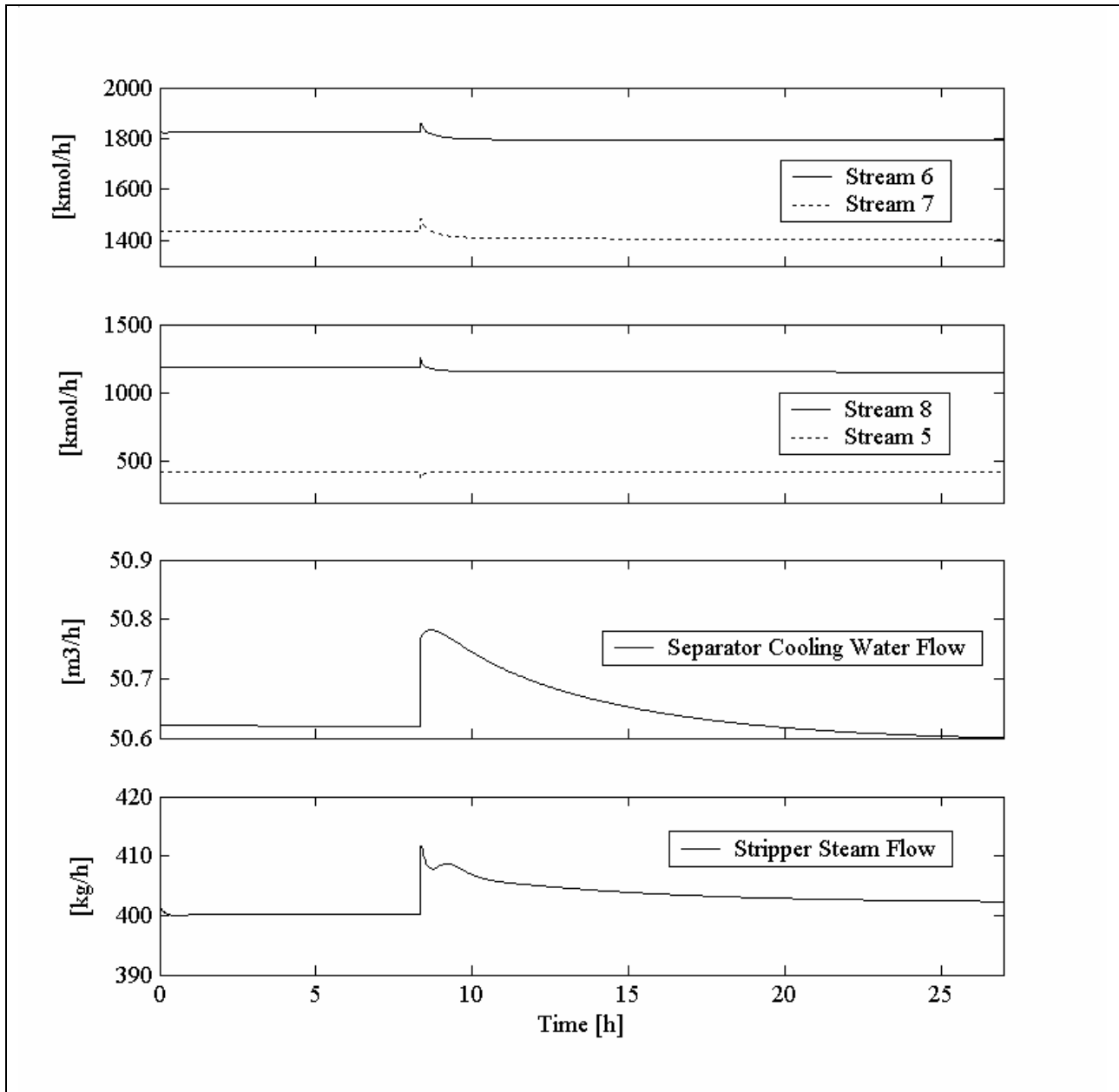
**Figure 12** Input and output stream trajectories of setpoint change study 2



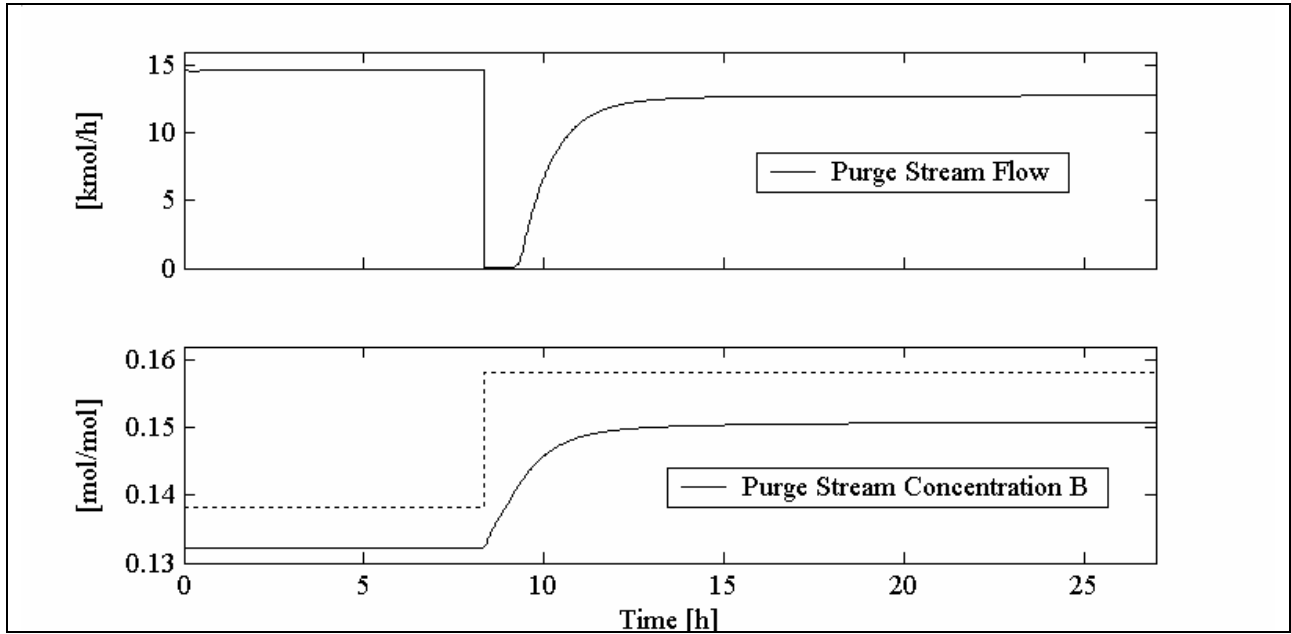
**Figure 13** Reactor temperature and cooling water flow, setpoint change study 2



**Figure 14** Reaction rates and reactor liquid level, setpoint change study 2



**Figure 15** Internal process streams, separator cooling water flow and stripper steam flow in setpoint change study 2



**Figure 16** Purge stream flow rate and concentration B, setpoint change study 3

## 10 List of Tables

Table 1 Number of Iterations, IPOPT, base case optimization .....	34
Table 2 Process manipulated variables .....	34
Table 3 Process operating constraints .....	34
Table 4 Composition of feed stream .....	35
Table 5 DAE model size and NLP system size after full discretization .....	35
Table 6 Setpoint changes for the base case (Downs and Vogel, 1993) .....	35

**Table 1** Number of Iterations, IPOPT, base case optimization

Initial Barrier Parameter $\mu$	$10^{-1}$	$10^{-2}$	$10^{-3}$	$10^{-4}$	$10^{-5}$	$10^{-6}$	$10^{-7}$	$10^{-8}$	$10^{-9}$
<b>Iterations needed</b>	31	23	17	13	15	14	16	17	30

**Table 2** Process manipulated variables

No	Name	Base case value	Low limit	High limit	Unit
1	A feed (stream 1)	11.2	0		kmol/h
2	D feed (stream 2)	114.5	0		kmol/h
3	E feed (stream 3)	98.0	0		kmol/h
4	A and C feed (stream 4)	417.5	0		kmol/h
5	Compressor recycle valve	22.210	0	100	%
6	Purge valve (stream 9)	40.064	0	100	%
7	Separator underflow (stream 10)	259.5	0		kmol/h
8	Stripper underflow (stream 11)	211.3	0		kmol/h
9	Reactor cooling water flow	93.37	0	227.1	m <sup>3</sup> /h
10	Condenser cooling water flow	49.37	0	272.6	m <sup>3</sup> /h
11	Stripper steam valve	47.446	0	100	%

**Table 3** Process operating constraints

Process variable	Normal operating limits		Shut down limits	
	low limit	high limit	Low limit	high limit
Reactor pressure	None	2895 kPa	none	3000 kPa
Reactor level	11.8 m <sup>3</sup>	21.3 m <sup>3</sup>	2.0 m <sup>3</sup>	24.0 m <sup>3</sup>
Reactor temperature	None	423 K	none	448 K
Separator level	3.3 m <sup>3</sup>	9m <sup>3</sup>	1.0 m <sup>3</sup>	12.0 m <sup>3</sup>
Stripper base level	3.5 m <sup>3</sup>	6.6 m <sup>3</sup>	1.0 m <sup>3</sup>	8.0 m <sup>3</sup>

**Table 4** Composition of feed stream

Stream	A (mol%)	B (mol%)	C (mol%)	D (mol%)	E (mol%)	F (mol%)
1	99.99	0.01	0.00	0.00	0.00	0.00
2	0.00	0.01	0.00	99.99	0.00	0.00
3	0.00	0.00	0.00	0.00	99.99	0.01
4	48.50	0.50	51.00	0.00	0.00	0.00

**Table 5** DAE model size and NLP system size after full discretization

DAE Model		NLP Optimization problem	
Number of differential equations	30	Number of variables	11400
Number of algebraic variables	160	Number of constraints	10740
Number of algebraic equations	149	Number of lower bounds	780
Difference (control variables)	11	Number of upper bounds	540
Number of time invariant parameters	0	Number of nonzeros in Jacobian	50550
Number of final time constraints	0	Number of nonzeros in Hessian	15180

**Table 6** Setpoint changes for the base case (Downs and Vogel, 1993)

Process variable	Type	Magnitude
Production rate change	Step	-15% Make a step change to the variable(s) used to set the process production rate so that the product flow leaving the stripper column base changes from 14,228 to 12,094 kg h <sup>-1</sup>
Reactor operating pressure change	Step	-60 kPa Make a step change so that the reactor operating pressure changes from 2805 to 2745 kPa
Purge gas composition of component B change	Step	+2% Make a step change so that the composition of component B in the gas purge changes from 13.82 to 15.82%