



Advanced step nonlinear model predictive control for air separation units

Rui Huang, Victor M. Zavala, Lorenz T. Biegler*

*Collaboratory for Process and Dynamic Systems Research, National Energy Technology Laboratory, P.O. Box 880, Morgantown, WV 26507-0880, USA
Department of Chemical Engineering, Carnegie Mellon University, 5000 Forbes Avenue, Pittsburgh, PA 15213, USA*

ARTICLE INFO

Article history:

Received 4 May 2008

Received in revised form 16 July 2008

Accepted 17 July 2008

Keywords:

Air separation units

Large-scale

Nonlinear model predictive control

Nonlinear programming

Advanced step NMPC

ABSTRACT

Cryogenic air separation units constitute an integral part of many industrial processes and next generation power plants. These units are characterized by fluctuating operating conditions to respond to changing product demands. The dynamics of these transitions are highly nonlinear and energy-intensive. Consequently, nonlinear model predictive control (NMPC) based on rigorous dynamic models is essential for high performance in these applications. Currently, the implementation of NMPC controllers is limited by the computational complexity of the associated on-line optimization problems. In this work, we make use of the so-called advanced step NMPC controller to overcome these limitations. We demonstrate that this sensitivity-based strategy reduces the on-line computational time to just a single CPU second, while incorporating a highly detailed dynamic air separation unit model. Finally, we demonstrate that the controller can handle nonlinear dynamics over a wide range of operating conditions.

© 2008 Elsevier Ltd. All rights reserved.

1. Introduction

Air separation units (ASUs) are cryogenic distillation systems that produce high purity nitrogen, oxygen and argon. Due to the high demand of these commodity materials, the ASU has become a crucial technology in many processes including next generation power plants. These units involve single or multiple energy-intensive cryogenic columns running at extremely low temperatures (−170 to −195 °C). As a consequence, the required degree of energy integration in these systems is quite high, which makes them difficult to operate. In addition, as the product demand fluctuates significantly in most ASU processes, the operating conditions need to be switched frequently. This switching leads to long transients that disrupt the system performance and profitability [25,27]. As expected, there is significant economic interest in reducing the operating costs of ASUs through advanced process control technology.

So far, the dominating control practice in ASU processes has been to adapt traditional regulatory controllers to maintain good performance. Today's trend has shifted towards multivariable control strategies such as model predictive control (MPC) [21,25]. MPC is a class of optimization-based control algorithms that uses model predictions to compute optimal future sequences of manipulated variables in order to maximize some measure of the plant performance [7,18]. MPC provides notable advantages over regulatory

control such as the direct handling of multivariable interactions and operational constraints. In this context, the use of input–output empirical models in MPC has been the dominant practice. This is due to the fact that this strategy avoids the development of rigorous dynamic models and the computational complexity of the associated quadratic programming problems is manageable. In fact, low-dimensional parametric MPC approaches have recently been used to reduce the on-line computational effort and have also been applied to ASUs [16]. Despite the success of linear MPC strategies, it is clear that their applicability to dynamic processes operating over wide operating regions would be strongly affected by the limited predictive capabilities of linear input–output *empirical* models.

In the last few years, there has been an increasing interest to apply nonlinear model predictive control (NMPC) with rigorous dynamic models in ASU processes. In previous studies, Roffel et al. [21] showed that multivariable control based on ASU rigorous models can provide significantly better performance than a traditional regulatory controller. In addition, Chen et al. [8] showed that NMPC can handle aggressive changes in the operating conditions of the ASU efficiently. Another advantage of NMPC is that complex economic objectives (not restricted to quadratic forms) can be directly incorporated in the optimization formulation, providing the controller with real-time optimization capabilities [15]. Despite these notable advantages, an important obstacle blocking the widespread use of NMPC is the computational complexity of the associated rigorous dynamic models, which comprise large sets of highly nonlinear differential and algebraic equations (DAEs).

Recent developments in large scale NLP algorithms and dynamic optimization strategies have enabled NMPC to become an

* Corresponding author. Address: Department of Chemical Engineering, Carnegie Mellon University, 5000 Forbes Avenue, Pittsburgh, PA 15213, USA. Tel.: +1 412 268 2232; fax: +1 412 268 7139.

E-mail address: Biegler@cmu.edu (L.T. Biegler).

attractive alternative in several industrial processes, and successful applications have been reported in recent years [1,12,17]. However, as larger and more sophisticated process models are considered, the computational complexity becomes an issue. This complexity translates into long sampling times and feedback delays that degrade controller performance and stabilizing properties [11,22].

In the context of ASU, Chen et al. [8] have addressed the on-line computational expense of NMPC through the development of models of reduced complexity based on compartmentalization concepts. In contrast, this study applies recently developed optimization algorithms and *on-line synchronization* strategies that enable the incorporation of rigorous, first principle, dynamic ASU models in NMPC. Such models are being validated and becoming more widespread in industry, and they are currently very useful for off-line simulation and optimization studies. In this study, we will incorporate these detailed, nonlinear models and study the performance of the so-called advanced step NMPC controller (*asNMPC*) [28]. The controller is based on a separation principle between background and inexpensive on-line computational tasks [9]. The controller exploits the predictive capabilities of the rigorous dynamic model to predict the future state of the plant and solve a predicted problem in background (between sampling times). Once the true state is revealed at the next sampling time, the controller responds to the inherent model errors and/or external disturbances through a fast on-line correction of the predicted solution. We demonstrate that the *asNMPC* controller is able to reduce the on-line feedback delay by over two orders of magnitude. In addition, the strategy makes use of simultaneous collocation-based formulations and a sparsity-exploiting interior-point solver to obtain fast background solutions. Finally, we compare the performance of the NMPC controller against that of a linear receding horizon controller which incorporates a fixed input–output empirical model. Several simulated operating scenarios arising in ASU processes are analyzed.

The paper is organized as follows. Section 2 discusses the structure of the ASU process under study and describes the rigorous dynamic model. Section 3 describes the dynamic optimization strategy, NLP algorithm and sensitivity concepts used to build the *asNMPC* controller. Section 4 presents three detailed case studies to illustrate the performance of the different control strategies. The last section concludes the paper and discusses topics for future work.

2. Air separation model

2.1. Process description

In this study, we consider an air separation unit that produces nitrogen with at least 99.9% purity, and oxygen with at least 96% purity. The impurity associated with the oxygen is argon. The specifics of the ASU process under study were reported in [4,27]. Here, ambient air is compressed in a large multistage compressor with intercooling followed by removal of water, carbon dioxide and hydrocarbons and by cooling in a multistream heat exchanger. As sketched in Fig. 1, this air feed mixture of oxygen, nitrogen and argon is then split into two substreams. The first stream consists of main air feed entering the bottom of the high pressure column (MA) and the second one consists of expanded air entering the 20th tray of the low pressure column (EA). Crude nitrogen gas (GN) from the main heat exchanger is also added to the 25th tray of the high pressure column. The high pressure column (bottom) contains 40 trays and operates at 5–6 bars, while the low pressure column (top) operates at 1–1.5 bars and also contains 40 trays. The reboiler of the low pressure column is integrated with the con-

denser of the high pressure column. The main products of the high pressure column are pure nitrogen (PNI) (>99.99%) and crude liquid oxygen (~50%). The crude oxygen stream is fed into the 19th tray of the low pressure column. In addition, an intermediate side stream from the 15th tray of the high pressure column (LN) is fed to the top of the low pressure column. A high purity separation is achieved in the low pressure column, leading to nitrogen gas with ~99% purity and oxygen (POX) with ~97% purity as products.

2.2. Model equations

Mathematical modeling of dynamic distillation columns is a well studied area [2,4,9,14,19,21]. Here, a detailed ASU model is derived under the following simplifying assumptions:

- Negligible vapor holdups on each tray, as the dynamics of the vapor phase are much faster than that of liquid phase.
- Ideal vapor phases.
- Well-mixed entering streams.
- Constant pressure drop on each tray.
- Equilibrium stage model.

The ASU model is represented by tray-by-tray equations consisting of mass balances (overall and component balances of nitrogen, oxygen and argon), energy balances, phase equilibrium, hydraulic and summation equations.

Overall mass balance

$$\frac{dM_i}{dt} = L_{i-1} + V_{i+1} - L_i - V_i + F_i \quad (1)$$

where i is the index of each tray, starting from the top of the column. M_i is the liquid mole holdup ([mol]) on tray i , L_i and V_i are liquid and vapor molar flow rates, respectively and F_i is the molar feed ($\frac{\text{mol}}{\text{min}}$). If there is no feed to tray i , then $F_i = 0$. In this case, the only nonzero values of F_i are those corresponding to expanded air (EA, U_1), main air feed (MA, U_2), liquid nitrogen (LN, U_3), crude gas nitrogen (GN, U_4) and crude oxygen stream as shown in Fig. 1.

Component balance

$$\frac{d(M_i x_{ij})}{dt} = L_{i-1} x_{i-1,j} + V_{i+1} y_{i+1,j} - L_i x_{ij} - V_i y_{ij} + F_i x_{ij}^f \quad (2)$$

where $j \in \text{COMP}$ is the index of each component, x_{ij} and y_{ij} are component mole fractions in the liquid and vapor phases, x_{ij}^f are the mole fractions of the feed. Alternatively, we can rewrite Eq. (2) using Eq. (1) as follows:

$$M_i \frac{dx_{ij}}{dt} = L_{i-1}(x_{i-1,j} - x_{ij}) + V_{i+1}(y_{i+1,j} - x_{ij}) - V_i(y_{ij} - x_{ij}) + F_i(x_{ij}^f - x_{ij}) \quad (3)$$

Energy balance

$$\frac{d(M_i h_i^l)}{dt} = L_{i-1} h_{i-1}^l + V_{i+1} h_{i+1}^v - L_i h_i^l - V_i h_i^v + F_i h_i^f \quad (4)$$

where $h_i^l = f^{hl}(T_i, P_i)$ and $h_i^v = f^{hv}(T_i, P_i)$ are liquid and vapor enthalpies in $\frac{\text{kJ}}{\text{mol}}$, and h_i^f is the feed enthalpy. Expressions and data to compute h_i^l and h_i^v can be found in a number of standard references. For this study, we use the information in [20]. Using Eq. (1), the energy balance (4) can be rewritten as

$$M_i \frac{dh_i^l}{dt} = L_{i-1}(h_{i-1}^l - h_i^l) + V_{i+1}(h_{i+1}^v - h_i^l) - V_i(h_i^v - h_i^l) + F_i(h_i^f - h_i^l) \quad (5)$$

Summation equation

$$1 = \sum_{j \in \text{COMP}} y_{ij} \quad (6)$$

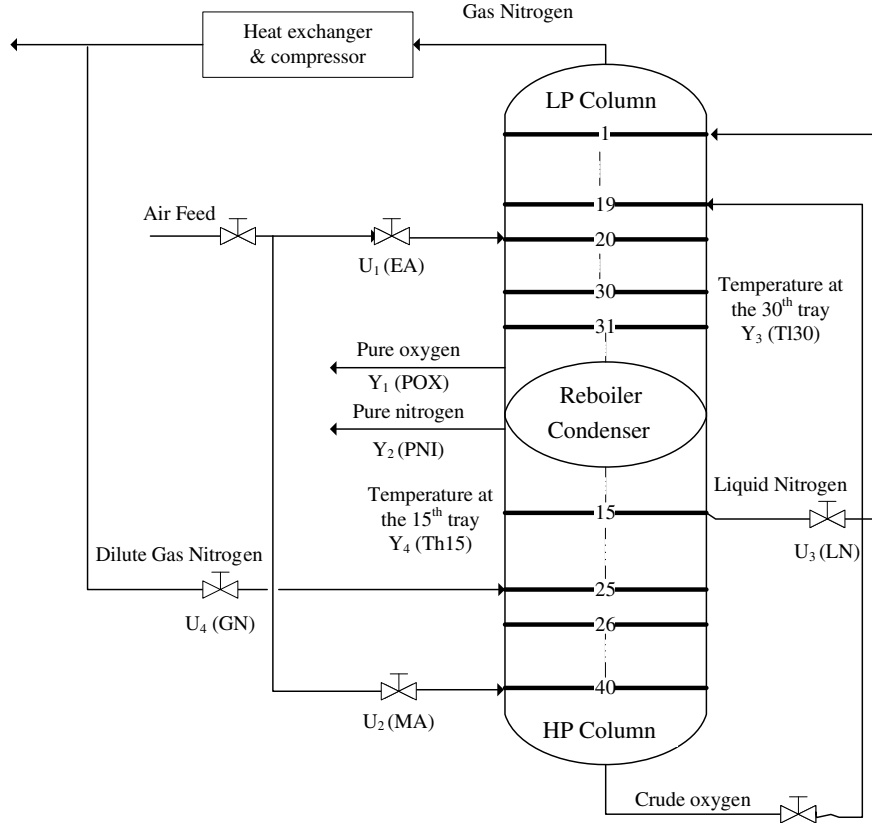


Fig. 1. Simplified flowsheet of ASU studied.

Hydraulic equation

$$L_i = k_d M_i \quad (7)$$

where $k_d = 0.5 \text{ min}^{-1}$ is a tuning constant determined from empirical data.

Vapor–liquid equilibrium

$$y_{ij} p_i = \gamma_{ij} x_{ij} p_{ij}^{\text{sat}} \quad (8)$$

where p_i is the total pressure on tray i , and $p_{ij}^{\text{sat}} = f_j^p(T_i)$ is the saturation pressure of pure component j on tray i . Expressions to compute p_{ij}^{sat} can be found in a number of standard references. For this study, we used the information in [20]. Symbol γ_{ij} denotes the liquid activity coefficient describing the non-ideal vapor–liquid equilibrium calculated from

$$\gamma_{i,N_2} = \exp \left[\frac{A_{N_2,O_2} x_{i,O_2}^2 + A_{N_2,Ar} x_{i,Ar}^2 + (A_{N_2,O_2} + A_{N_2,Ar} - A_{O_2,Ar}) x_{i,O_2} x_{i,Ar}}{RT_i} \right] \quad (9a)$$

$$\gamma_{i,O_2} = \exp \left[\frac{A_{N_2,O_2} x_{i,N_2}^2 + A_{O_2,Ar} x_{i,Ar}^2 + (A_{N_2,O_2} + A_{O_2,Ar} - A_{N_2,Ar}) x_{i,N_2} x_{i,Ar}}{RT_i} \right] \quad (9b)$$

$$\gamma_{i,Ar} = \exp \left[\frac{A_{N_2,Ar} x_{i,N_2}^2 + A_{O_2,Ar} x_{i,O_2}^2 + (A_{N_2,Ar} + A_{O_2,Ar} - A_{N_2,O_2}) x_{i,N_2} x_{i,O_2}}{RT_i} \right] \quad (9c)$$

Here R is the ideal gas constant and the coefficients $A_{j,k}$ account for the liquid phase interactions between components j and k . These can be calculated using the Margules equations as reported in [2].

Eqs. (1), (3), (5)–(9) lead to a differential algebraic equation (DAE) system, with the differential variables M_i , x_{ij} and h_i^t . Eliminating the dynamics of the vapor phase reduces the stiffness of the model but makes the DAE system index 2. In other words, we found that the algebraic variable V_i cannot be explicitly recovered from the algebraic equations.

Solving the index 2 DAE system is often difficult as consistent initial conditions need to be determined. In order to avoid this, the system was reduced to index 1 by differentiating the summation Eq. (6) with respect to time

$$0 = \sum_{j \in \text{COMP}} \frac{dy_{ij}}{dt} = \sum_{j \in \text{COMP}} \left[\frac{dK_{ij}}{dt} x_{ij} + K_{ij} \frac{dx_{ij}}{dt} \right] \quad (10)$$

where we define $K_{ij} = \gamma_{ij} p_{ij}^{\text{sat}} / p_i$ to simplify the notation. As a result, K_{ij} is a function of temperature T_i and component concentration in each tray x_{ij} . Applying the chain rule to (10), we obtain

$$0 = \sum_{j \in \text{COMP}} x_{ij} \left[\frac{\partial K_{ij}}{\partial T_i} \frac{dT_i}{dt} + \sum_{k \in \text{COMP}} \frac{\partial K_{ij}}{\partial x_{i,k}} \frac{dx_{i,k}}{dt} \right] + \sum_{j \in \text{COMP}} K_{ij} \frac{dx_{ij}}{dt} \quad (11)$$

By combining Eqs. (11) and (3) and introducing the dummy variables \bar{x}_{ij} and \bar{T}_i , we obtain

$$\bar{x}_{ij} := \frac{dx_{ij}}{dt} = \frac{L_{i-1}(x_{i-1,j} - x_{ij}) + V_{i+1}(y_{i+1,j} - x_{ij}) - V_i(y_{i,j} - x_{ij}) + F_i(x_{i,j}^f - x_{ij})}{M_i} \quad (12)$$

$$\bar{T}_i := \frac{dT_i}{dt} = - \frac{\sum_{j \in \text{COMP}} \left[x_{ij} \sum_{k \in \text{COMP}} \left(\frac{\partial K_{ij}}{\partial x_{i,k}} \bar{x}_{i,k} \right) + K_{ij} \bar{x}_{i,k} \right]}{\sum_{j \in \text{COMP}} x_{ij} \partial K_{ij} / \partial T_i} \quad (13)$$

Note that by changing the left hand side of the energy balance (5) we can rewrite this as an algebraic equation in terms of the dummy variables

$$M_i \left(\frac{\partial h_i^L}{\partial T_i} \bar{T}_i + \sum_{j \in \text{COMP}} \frac{\partial h_i^L}{\partial x_{ij}} \bar{x}_{ij} \right) = L_{i-1}(h_{i-1}^L - h_i^L) + V_{i+1}(h_{i+1}^V - h_i^L) - V_i(h_i^V - h_i^L) + F_i(h_i^f - h_i^L) \quad (14)$$

The reformulated index 1 DAE system now consists of Eqs. (1), (3), (6)–(12), (13), (14) for each tray and component. The ASU model contains 320 differential equations and 1200 algebraic equations with M_i and x_{ij} as differential variables.

3. The advanced-step NMPC controller

In this section, we describe the basic components of the advanced-step NMPC controller. We first discuss the dynamic optimization problem formulation and the solution strategy. Then, we make use of the parametric properties of the moving-horizon problems arising in NMPC to derive a strategy able to reduce the on-line computational delay of NMPC.

3.1. Dynamic optimization strategy

We consider a dynamic optimization problem with initial conditions given by the current process state $x(k)$ at sampling time t_k

$$\min_{u(t)} \int_{t_k}^{t_{k+N}} \psi(z(t), y(t), u(t)) dt \quad (15a)$$

$$\text{s.t.} \quad \frac{dz(t)}{dt} = f(z(t), y(t), u(t)) \quad (15b)$$

$$0 = g(z(t), y(t), u(t)) \quad (15c)$$

$$z(t_k) = x(k) \quad (15d)$$

$$z^L \leq z(t) \leq z^U \quad (15e)$$

$$y^L \leq y(t) \leq y^U \quad (15f)$$

$$u^L \leq u(t) \leq u^U \quad (15g)$$

$$t \in [t_k, t_{k+N}] \quad (15h)$$

where t is the scalar time dimension, N is the horizon length, $z(t) \in \mathbb{R}^{n_z}$ is a vector of dynamic state variables, $y(t) \in \mathbb{R}^{n_y}$ is a vector of algebraic variables and $u(t) \in \mathbb{R}^{n_u}$ is a vector of manipulated variables. The DAE system (15b)–(15d) contains the rigorous description of the plant. Eqs. (15e)–(15g) are bound constraints for the differential and algebraic states and for the manipulated variables, respectively. Finally, note that the dynamic optimization problem can be seen as parametric in the initial state $x(k)$, $k > 1$.

For the solution of large-scale dynamic optimization problems, direct methods based on nonlinear programming (NLP) strategies are particularly efficient. Among these, single shooting [23,24], multiple shooting [5] and simultaneous collocation-based [3] approaches are the alternatives. In this work, we follow a simultaneous collocation-based approach. In this approach, the infinite-dimensional representation of problem (15) is transcribed into a finite one through discretization schemes. For accurate approximations with guaranteed convergence properties, we use a finite element discretization at Radau collocation points [3,13]. The discretization procedure leads to a large-scale NLP problem which directly incorporates the discretized dynamic model as algebraic constraints. Important advantages of this method involve the ability to exploit the sparse structure of the dynamic model directly in the NLP solver, which tends to reduce the overall computational complexity. In addition, since the dynamic model is in completely algebraic form, cheap first and second order derivative information can be obtained directly from modeling platforms. On the other

hand, this approach requires efficient NLP solvers that are able to exploit the sparsity of the problem and to handle the nonlinearity and complexity of the large-scale dynamic model. In this work, we make use of the interior-point solver IPOPT [26] to solve the resulting NLP problem. In the following subsection we describe the basics of the IPOPT algorithm and the recently incorporated NLP sensitivity capabilities. These components enable the computation of fast approximate solutions for parametric dynamic optimization problems arising in NMPC.

3.2. IPOPT algorithm and NLP sensitivity

After discretization, problem (15) can be stated as a general parametric NLP problem of the form

$$\min_{w(p)} F(w(p), p) \quad (16a)$$

$$\text{s.t.} \quad c(w(p), p) = 0, \quad w(p) \geq 0 \quad (16b)$$

with variables w , objective and constraint functions F and c , respectively, and parameters p . Notice the implicit dependence of the problem variables on the particular value of the parameter. In the context of NMPC, this parameter can be denoted as the initial conditions $x(k)$. IPOPT handles the bound constraints implicitly through logarithmic barrier terms added to the objective function,

$$\min_{w(p)} F(w(p), p) - \mu \sum_{i=1}^n \ln(w^{(i)}(p)) \quad (17a)$$

$$\text{s.t.} \quad c(w(p), p) = 0 \quad (17b)$$

where $\mu > 0$ is a barrier parameter. Symbol $w^{(i)}(p)$ denotes the i th component of vector $w(p)$. The solution of the barrier problem (17) converges to the solution of the original NLP (16) as the barrier parameter tends to zero.

To solve each barrier problem, IPOPT applies Newton's method to the KKT conditions of system (17). This results in the following large-scale linear system solved at each iteration j

$$\begin{bmatrix} H_j & A_j & -I \\ A_j^T & 0 & 0 \\ V_j & 0 & Z_j \end{bmatrix} \begin{bmatrix} \Delta w \\ \Delta \lambda \\ \Delta v \end{bmatrix} = - \begin{bmatrix} \nabla F(w_j(p), p) + A_j \lambda_j(p) - v_j(p) \\ c(w_j(p), p) \\ Z_j V_j e - \mu_j e \end{bmatrix} \quad (18)$$

where λ and v are the Lagrange multipliers for the equality constraints and bounds, respectively. In addition, $Z := \text{diag}(w(p))$, $V := \text{diag}(v(p))$, $H := H(w(p), p)$, is the Hessian of the Lagrange function $\mathcal{L} = F(w(p), p) + c(w(p), p)^T \lambda(p) - v(p)^T w(p)$, and $A := A(w(p), p)$ is the constraint Jacobian. It is important to emphasize that the most expensive step at each iteration in the algorithm is the factorization of the KKT matrix in the left hand side of Eq. (18). Depending on the size and structure of the problem, this factorization step can take a significant amount of computation. In this work, we provide the IPOPT solver with exact Hessian and Jacobian information through the modeling platform AMPL. This accelerates the local convergence of Newton's method and allows efficient solution of NLPs with many degrees of freedom.

Now consider the solution of the NLP (16) for a given nominal parameter vector $p = p_0$ where the KKT system (18) can be expressed in condensed form as

$$\mathbf{K}^*(p_0) \Delta s = -\varphi(s^*(p_0), p_0) \quad (19)$$

where $s^*(p_0)^T = [w^*(p_0)^T, \lambda^*(p_0)^T, v^*(p_0)^T]$ is the optimal triplet vector while $\varphi(\cdot, \cdot)$ and $\mathbf{K}^*(\cdot)$ are the KKT conditions and KKT matrix evaluated at this solution, respectively.

We further assume that the objective and constraints in NLP (16) are at least twice continuously differentiable in p and that a given nominal solution $s^*(p_0)$ satisfies the linear independence constraint qualification (LICQ), the sufficient second order

conditions (SSOC) and strict complementary slackness. If the parameter vector p_0 enters linearly in the objective function and constraints (e.g., as the initial conditions in the dynamic optimization problem), it is possible to show [10,28] that replacing the nominal parameter p_0 for the perturbed parameter p in (19) leads to the first order step $\Delta s = \tilde{s}(p) - s^*(p_0)$. Here, $\tilde{s}(p)$ is an approximation to the true optimal solution $s^*(p)$ satisfying

$$|\tilde{s}(p) - s^*(p)| \leq L|p - p_0|^2 \quad (20)$$

where L is a positive Lipschitz constant. The above result has important practical implications since the factorization of the KKT matrix in (19) is already available as a natural outcome of the NLP solver. As a consequence, the second order approximate solution can be obtained through a single backsolve which can be performed at least an order of magnitude faster than the solution of the full NLP problem.

3.3. Control algorithm

The *asNMPC* control algorithm exploits the parametric properties of the dynamic optimization problem in NMPC and the predictive capabilities of the rigorous dynamic model. The algorithm can be summarized as:

- (1) *Background calculation*: Having $x(k)$ and $u(k)$ at t_k , predict the future state of the system $z(t_{k+1})$ using the dynamic model. Assuming the computations can be completed within one sampling time, solve the NLP based on (15) over $t \in [t_{k+1}, t_{k+N+1}]$ with $p_0 = z(t_{k+1})$.
- (2) *On-line update*: At t_{k+1} , obtain the true state $x(k+1)$. Set $p = x(k+1)$ and use the sensitivity equation, $\mathbf{K}^*(p_0)\Delta s = -\varphi(s^*(p_0), p)$, to get the fast updated solution. Extract the control action $u(k+1)$ from the approximate solution vector and inject to the plant.
- (3) *Iterate*: Set $k \leftarrow k+1$ and go to background.

The *asNMPC* controller presents an attractive alternative to approximate the performance of a hypothetical *ideal* NMPC strategy that provides optimal instantaneous feedback at each sampling time (i.e. no computational delay). Using NLP sensitivity theory, it is possible to bound the approximation error of the *asNMPC* controller in a rigorous manner. In addition, the controller enjoys the same nominal stability properties of the ideal NMPC, and its robust stability margins can be bounded by the uncertainty description implicit in the perturbation $|x(k) - z(t_k)|$ [28].

4. NMPC case studies for ASU

In this section, we evaluate the performance of the *asNMPC* controller in ASU with changing production demands. To design the controller, we choose the molar flow rate of pure oxygen (POX- Y_1) and the molar flow rate of pure nitrogen (PNI- Y_2) as output variables. The objective is to force the outputs to follow their set-points while satisfying purity requirements.

Current ASU technology allows product composition to be measured directly. Moreover, nonlinear transformation of these measurements are commonly used for linear MPC controllers. On the other hand, we prefer to use the tray temperatures because it is a cheap, continuous indirect composition measurement common to distillation control, and we can also choose more sensitive intermediate tray temperatures that add more robustness to the controller. In this study, temperatures at several sensitive trays are chosen as output variables. In particular, we choose the temperature at 30th tray in the low pressure column (T130- Y_3), and temperature at the 15th tray in the high pressure column (Th15- Y_4).

Four stream flow rates are considered as manipulated variables. This includes the expanded air feed (EA- U_1), main air feed (MA- U_2), reflux liquid nitrogen (LN- U_3) and crude gas nitrogen (GN- U_4). All set-points and reference values for the manipulated variables were determined through steady-state simulations with AspenPlus.

The objective function for the NMPC controller is in the following form:

$$\text{Min}_{u(t)} \int_{t_k}^{t_{k+N}} \left((y(t) - y_{\text{set}})^T \mathcal{W} (y(t) - y_{\text{set}}) + (u(t) - u_{\text{ref}})^T \mathcal{U} (u(t) - u_{\text{ref}}) \right) dt \quad (21)$$

where $y(t)$ is a vector of controlled variables, $u(t)$ is a vector of manipulated variables and y_{set} and u_{ref} are the set-points and reference values for the output and manipulated variables, respectively. The symbols \mathcal{W} and \mathcal{U} denote diagonal weighting matrices. The diagonal element in \mathcal{W} corresponding to each manipulated variable is set to 1×10^{-5} except the one corresponding to EA- U_1 , which is set to 1×10^{-4} . The diagonal elements in \mathcal{U} corresponding to T130 and Th15 are set to 3×10^{-2} while the elements corresponding to POX and PNI are 1×10^{-4} .

We consider three case studies to demonstrate the performance of the *asNMPC* controller. The first case considers *ramp* changes of the production rate set-point. Here, we contrast the performance of *asNMPC* controller against that of a receding horizon controller that uses a fixed linearized dynamic process model. With this, we demonstrate that, despite the error introduced by NLP sensitivity approximations, the *asNMPC* controller can still handle the nonlinear process dynamics over a wide range of operating conditions. The second case considers *step* changes of the production rate set-point in the presence of random disturbances. Here, we compare the performance of *asNMPC* with that of a hypothetical, ideal NMPC controller (without computational delay). Finally, we compare the performance of these two controllers in a third case considering larger ramps of the production rates.

4.1. Case 1: ramping the product streams

In this scenario, the ASU starts from a nominal steady-state computed from simulation. The oxygen and nitrogen production rate set-points (and associated reference values for the manipulated variables) are reduced by 30% through a ramp change from $t = 30$ to $t = 60$ min. After this, they undergo a ramp increase from $t = 1000$ to $t = 1030$ back to their original values. For this simulation, we assume that the model is perfect and no unmeasured disturbances are present. The control horizon is chosen to be equal to the prediction horizon with a total length of 100 min distributed over 20 finite elements. The sampling time is set to 5 min. After full discretization of the dynamic optimization problem using collocation on finite elements, the resulting NLP contains 117,140 variables and 116,900 constraints. The NLP is very sparse with up to 5 nonzero entries per row in the Jacobian, and 4 nonzero entries per row in the Hessian.

A total of 400 moving horizon optimization problems was solved. We set the initial barrier parameter to $\mu = 10^{-7}$ in order to promote fast convergence of the solver. The solver took 5–8 iterations and 120–220 CPU seconds to converge with a tight tolerance of 1×10^{-6} . All simulations were performed on an Intel DuoCore 2.4 GHz personal computer. Based on these computational times, it is clear that a conventional implementation of NMPC would introduce a feedback delay of almost 4 min. The *asNMPC* controller brings the on-line computational time down to 1 CPU second, corresponding to the time required to perform a single backsolve with the fixed KKT matrix. This on-line computational time is over 100 times less than the full solution of the NLP.

The closed-loop profiles for the first case study are shown in Figs. 2 and 3. Note that the *asNMPC* controller is able to track the production rate ramps well while maintaining the tray temperatures close to their set-points (compare the dashed and dot-dashed lines). To illustrate the benefits of the NMPC controller and its nonlinear dynamic model, we considered the same controller but with a linear dynamic model. Here the linear process model is derived by identifying the data generated by dynamic simulation of the rigorous nonlinear model using MATLAB. The linear Multiple Input Multiple Output (MIMO) model is then obtained by combining all the Multiple Input Single Output (MISO) ARX models between inputs and outputs. Each ARX model is chosen with a structure that achieves the smallest Akaike Information Criteria (AIC) [6]. In order to assess the impact of the nonlinear model within the NMPC controller, the prediction horizon and sampling times are also set to 100 min and 5 min, respectively.

From Figs. 2 and 3 note that the *asNMPC* controller keeps the controlled variables close to their set-points. On the other hand, the controller with the linear model presents large deviations while *asNMPC* recovers quickly. Fig. 3 shows that the manipulated variables of the linear controller are aggressive and tend to oscillate, while their *asNMPC* counterparts stay close to their reference values. This is mainly due to the fact that the *asNMPC* controller can still handle the nonlinear process dynamics due to the background update of the KKT matrix at each time step.

4.2. Case 2: step change in product streams

The first case demonstrates that *asNMPC* with a rigorous nonlinear model is expected to have better performance over a wider range than a controller based on a fixed linear model. In the second case study, the set-points of the controlled variables undergo a more aggressive step change. The profiles are presented in Figs. 4 and 5. At $t = 30$, the set-points and reference values are instantaneously reduced by 30%, and later increased back to their original values at $t = 1000$. In addition, a 5% random disturbance is added to the molar holdups to test the robustness of the controllers. A total of 400 moving horizon problems was solved. For the same barrier parameter μ and tolerance levels, IPOPT takes up to 10 iterations and 240 CPU seconds to converge the NLPs. The time required for the on-line calculation of the *asNMPC* controller was still around 1 CPU second. In addition, it is worth mentioning that our

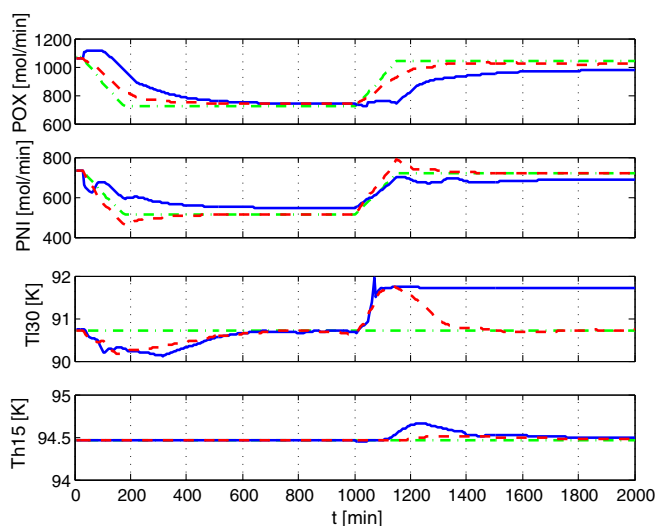


Fig. 2. Controlled variables for case one. The dot-dashed line is the set-point, the solid line is the linear controller profile and the dashed line is the *asNMPC* profile.

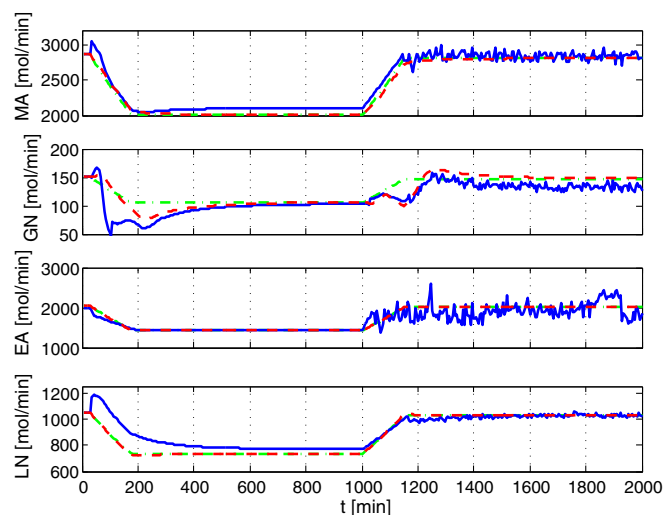


Fig. 3. Manipulated variables for case one. The dot-dashed line is the reference value, the solid line is the linear controller profile and the dashed line is the *asNMPC* profile.

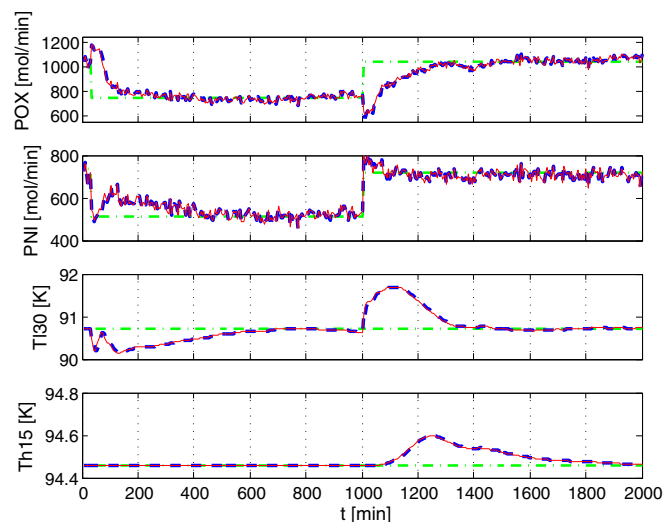


Fig. 4. Controlled variables for case two. The dot-dashed line is the set-point, the thin solid line is the ideal NMPC profile and the dashed line is the *asNMPC* profile.

linear controller with the same tuning parameters tends to be unstable for this case.

In this case study, we also compare the performance of the *asNMPC* controller with that of the hypothetical ideal NMPC controller. As shown in Figs. 4 and 5, the *asNMPC* controller is able to reject the random disturbances during both transients. Moreover, note from Fig. 6 that the product purities are satisfied during the entire time frame even though they are controlled indirectly through the tray temperatures. Compared to the product purities, the temperature profiles seem to show larger deviations due to the more aggressive step change. Finally, it is clear that the *asNMPC* controller yields very good sensitivity approximations despite relatively large disturbances.

4.3. Case 3: large ramps in product streams

The ramp change strategy helped us to consider wider transitions. For the third case study, the set-points of the production rates undergo a ramp reduction from $t = 30$ to $t = 60$ by 40%, along

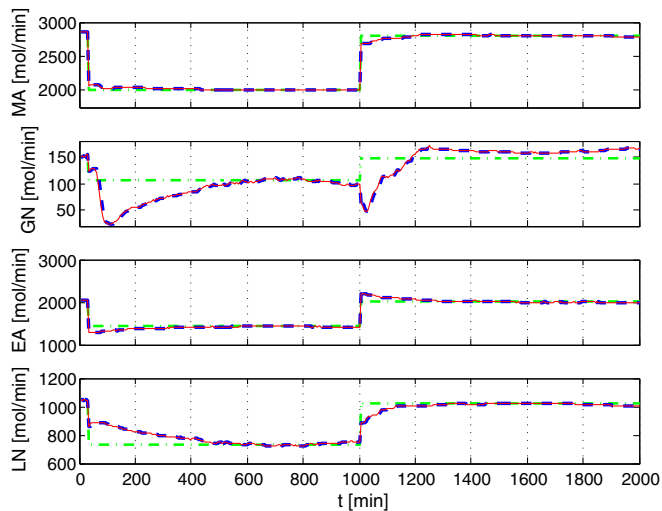


Fig. 5. Manipulated variables for case two. The dot-dashed line is the reference value, the thin solid line is the ideal NMPC profile and the dashed line is the *asNMPC* profile.

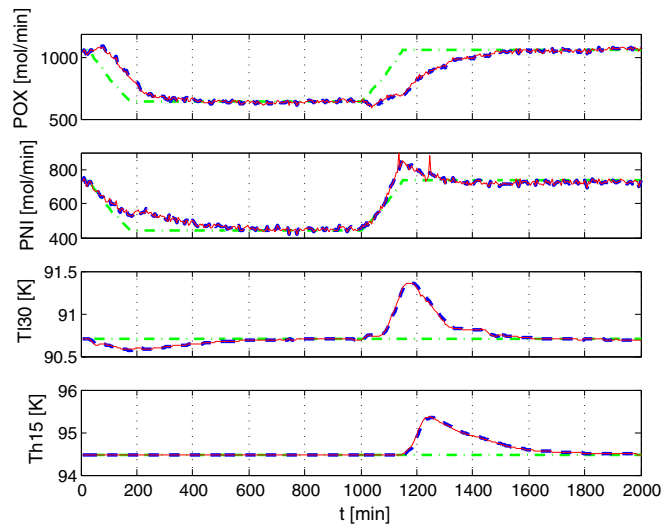


Fig. 7. Controlled variables of case three. The dot-dashed line is the set-point, the thin solid line is the ideal NMPC profile and the dashed line is the *asNMPC* profile.

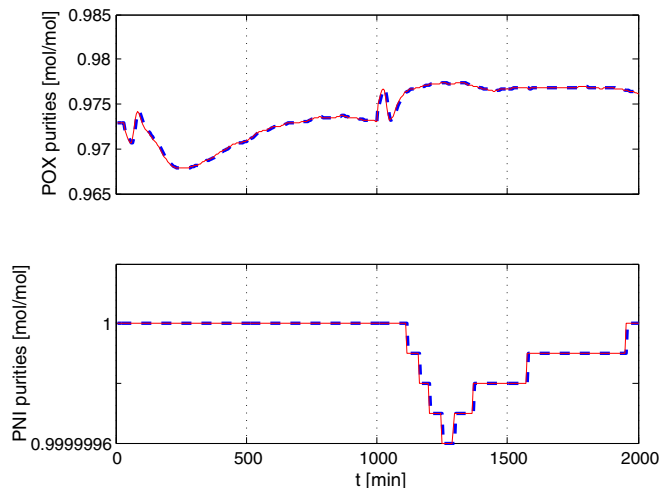


Fig. 6. Product purities of case two. The thin solid line is the ideal NMPC profile and the dashed line is the *asNMPC* profile.

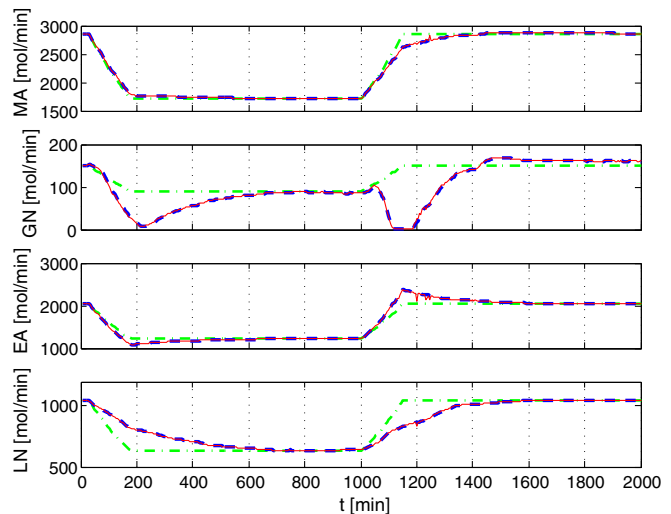


Fig. 8. Manipulated variables of case three. The dot-dashed line is the reference value, the thin solid line is the ideal NMPC profile and the dashed line is the *asNMPC* profile.

with a ramp back to their original values from $t = 1000$ to $t = 1030$. The results are presented in Figs. 7 and 8. As in the second case, a 5% random disturbance is added to the molar holdup on each tray. All the tuning parameters stay as before. Obviously, the production rates can easily be switched while the product purities are kept at target. Despite the large ramp change, IPOPT was able to solve each of the 400 moving horizon problems in around 8 iterations and 220 CPU seconds. Again, on-line computation required around 1 CPU seconds. Furthermore, the *asNMPC* controller still yields a nearly perfect approximation of the ideal NMPC controller performance as seen in Fig. 9.

5. Conclusion and future work

Air separation units are an essential technology in many industrial processes. The wide changes in production rates and tight energy integration encountered in these units demand advanced control technology for robust and profitable operation. In this context, NMPC using rigorous ASU models is an attractive candidate but is limited by the associated on-line computational effort. In

this work, we make use of the recently proposed advanced-step NMPC (*asNMPC*) algorithm to overcome these limitations. The proposed strategy is demonstrated using several scenarios arising in industrial ASU units. Here, we show that the sensitivity-based *asNMPC* controller can handle the nonlinear dynamics of the process over wide production rate changes. In addition, the *asNMPC* controller provides nearly identical performance to that of a hypothetical ideal NMPC controller with a minimum of on-line computational effort.

Future work will consider the extension of sensitivity-based NMPC controllers to deal with uncertainty and updating of the nonlinear dynamic model. Our current NMPC framework assumes that state estimates are available instantaneously and the dynamic model is perfect. In reality, state and parameter estimation need to be used, even though this would require additional computational effort, leading to even longer feedback delays. In order to overcome this delay, we plan to couple the MHE algorithm proposed in [29] to the *asNMPC* controller. This would allow states and model parameters to be updated with negligible on-line computation.

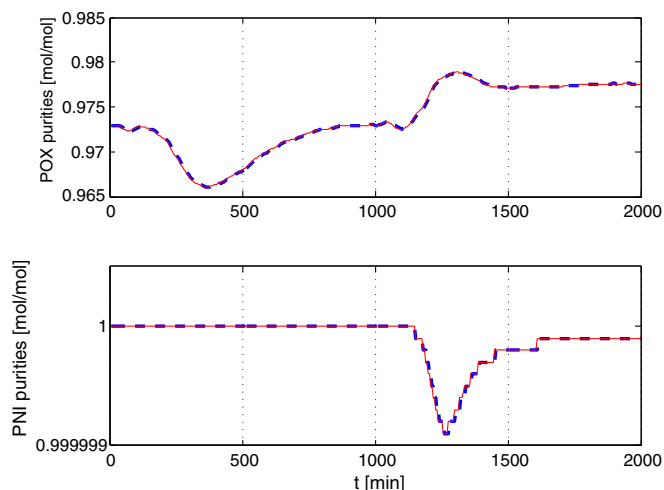


Fig. 9. Product purities of case three. The thin solid line is the ideal NMPC profile and the dashed line is the *asNMPC* profile.

Finally, with these developments, our ultimate goal is to consider the implementation of NMPC controllers with economic objectives able to minimize the operational costs of ASU units.

References

- [1] R.D. Bartusiak, NLMPC: a platform for optimal control of feed- or product-flexible manufacturing, in: R. Findeisen, F. Allgöwer, L.T. Biegler (Eds.), *Assessment and Future Directions of Nonlinear Model Predictive Control*, Springer, 2007, pp. 367–381.
- [2] S. Bian, S. Khowinij, M. Henson, P. Belanger, L. Megan, Compartmental modeling of high purity air separation columns, *Computers & Chemical Engineering* 29 (2005) 2096–2109.
- [3] L.T. Biegler, A.M. Cervantes, A. Wächter, Advances in simultaneous strategies for dynamic process optimization, *Chemical Engineering Science* 57 (2002) 575–593.
- [4] K. Bloss, *Dynamic Process Optimization through Adjoint Formulations and Constraint Aggregation*, Ph.D. Thesis, Lehigh University, 2000.
- [5] H.G. Bock (Ed.), *Numerical Treatment of Inverse Problems in Differential and Integral Equations: Proceedings of an International Workshop*, Heidelberg, Federal Republic of Germany, 1983.
- [6] G. Box, G.M. Jenkins, G.C. Reinsel, *Time Series Analysis: Forecasting and Control*, third ed., Prentice-Hall, 1994.
- [7] E.F. Camacho, C. Bordons, Nonlinear model predictive control: an introductory review, in: R. Findeisen, F. Allgöwer, L.T. Biegler (Eds.), *Assessment and Future Directions of Nonlinear Model Predictive Control*, Springer, 2007, pp. 1–16.
- [8] Z. Chen, M. Henson, P. Belanger, L. Megan, Nonlinear model predictive control of cryogenic air separation columns, in: *Proceedings of AIChE Annual Meeting*, 2007.
- [9] M. Diehl, I. Uslu, R. Findeisen, S. Schwarzkopf, F. Allgöwer, et al., Real-time optimization for large scale processes: nonlinear model predictive control of a high purity distillation column, *Journal of Process Control* 12 (2002) 577–585.
- [10] A.V. Fiacco, *Introduction to Sensitivity and Stability Analysis in Nonlinear Programming*, Academic Press, New York, 1983.
- [11] R. Findeisen, F. Allgöwer, Computational delay in nonlinear model predictive control, in: *Proceedings of the International Symposium on Advanced Control of Chemical Processes, ADCHEM'03*, 2003.
- [12] R. Franke, J. Engell, Integration of advanced model based control with industrial it, in: R. Findeisen, F. Allgöwer, L.T. Biegler (Eds.), *Assessment and Future Directions of Nonlinear Model Predictive Control*, Springer, 2007, pp. 399–406.
- [13] S. Kameswaran, L.T. Biegler, Convergence rates for direct transcription of optimal control problems using collocation at radau points, *Computational Optimization and Applications* 41 (2008) 81–126.
- [14] T. Kronseider, O.V. Stryk, R. Bulirsch, A. Kröner, Towards nonlinear model-based predictive optimal control of large-scale process models with application to air separation unit, in: M. Grötschel, S.O. Krumke (Eds.), *Online Optimization of Large Scale Systems: State of the Art*, Springer-Verlag, 2001, pp. 385–412.
- [15] A. Küpper, S. Engell, Nonlinear model predictive control of the hashimoto simulated moving bed process, in: R. Findeisen, F. Allgöwer, L. T Biegler (Eds.), *Assessment and Future Directions of Nonlinear Model Predictive Control*, Springer, 2007, pp. 473–483.
- [16] J. Mandler, N. Boziniis, V. Sakizlis, E. Pistikopoulos, A. Prentice, H. Ratna, R. Freeman, Parametric model predictive control of air separation, in: *IFAC Symposium on Advanced Control of Chemical Processes, ADCHEM, Gramado, Brazil*, 2006.
- [17] Z.K. Nagy, B. Mahn, R. Franke, F. Allgöwer, Real-time implementation of nonlinear model predictive control of batch process in an industrial framework, in: R. Findeisen, F. Allgöwer, L. T Biegler (Eds.), *Assessment and Future Directions of Nonlinear Model Predictive Control*, Springer, 2007, pp. 465–472.
- [18] S.J. Qin, T.A. Badgwell, A survey of industrial model predictive control technology, *Control Engineering Practice* 11 (2003) 733–764.
- [19] A. Raghunathan, M. Diaz, L. Biegler, An mpec formulation for dynamic optimization of distillation operations, *Computers & Chemical Engineering* (2004).
- [20] R.C. Reid, J.M. Prausnitz, B.E. Poling, *The Properties of Gases and Liquids*, McGraw-Hill, New York, 1987.
- [21] B. Roffel, B.H.L. Betlem, J.A.F. Ruijter, First principles dynamic modeling and multivariable control of a cryogenic distillation process, *Computers & Chemical Engineering* 24 (2000) 111–123.
- [22] L.O. Santos, P. Afonso, J. Castro, N. Oliveira, L.T. Biegler, Online implementation of nonlinear mpc: an experimental case study, *Control Engineering Practice* 9 (2001) 847–857.
- [23] V.S. Vassiliadis, R.W.H. Sargent, C.C. Pantelides, Solution of a class of multistage dynamic optimization problems. part one-algorithmic framework, *Industrial and Engineering Chemistry Research* 33 (1994) 2115–2123.
- [24] V.S. Vassiliadis, R.W.H. Sargent, C.C. Pantelides, Solution of a class of multistage dynamic optimization problems. Part two-problems with path constraints, *Industrial and Engineering Chemistry Research* 33 (1994) 2123–2133.
- [25] D. Vinson, Air separation control technology, *Computers & Chemical Engineering* 30 (2006) 1436–1446.
- [26] A. Wächter, L.T. Biegler, On the implementation of a primal–dual interior point filter line search algorithm for large-scale nonlinear programming, *Mathematical Programming* 106 (2006) 25–27.
- [27] V. White, J. Perkins, D. Espie, Switchability analysis, *Computers & Chemical Engineering* 20 (1996) 469–474.
- [28] V. Zavala, L. Biegler, The advanced step nmpc controller: optimality, stability and robustness, *Automatica*, in press.
- [29] V. Zavala, C. Laird, L. Biegler, A fast moving horizon estimation algorithm based on nonlinear programming sensitivity, *Journal of Process Control* 18 (2008) 876–884.