Computers and Chemical Engineering 33 (2009) 575–582

Contents lists available at ScienceDirect

Computers and Chemical Engineering

journal homepage: www.elsevier.com/locate/compchemeng

Large-scale nonlinear programming using IPOPT: An integrating framework for enterprise-wide dynamic optimization

L.T. Biegler*, V.M. Zavala

Chemical Engineering Department, Carnegie Mellon University, Pittsburgh, PA 15213, USA

ARTICLE INFO

Article history:
Received 21 March 2008
Received in revised form 20 August 2008
Accepted 21 August 2008
Available online 31 August 2008

Keywords:
Real-time optimization
Model predictive control
Moving horizon estimation
Nonlinear programming
Sensitivity

ABSTRACT

Integration of real-time optimization and control with higher level decision-making (scheduling and planning) is an essential goal for profitable operation in a highly competitive environment. While integrated large-scale optimization models have been formulated for this task, their size and complexity remains a challenge to many available optimization solvers. On the other hand, recent development of powerful, large-scale solvers leads to a reconsideration of these formulations, in particular, through development of efficient large-scale barrier methods for nonlinear programming (NLP). As a result, it is now realistic to solve NLPs on the order of a million variables, for instance, with the IPOPT algorithm. Moreover, the recent NLP sensitivity extension to IPOPT quickly computes approximate solutions of perturbed NLPs. This allows on-line computations to be drastically reduced, even when large nonlinear optimization models are considered. These developments are demonstrated on dynamic real-time optimization strategies that can be used to merge and replace the tasks of (steady-state) real-time optimization and (linear) model predictive control. We consider a recent case study of a low density polyethylene (LDPE) process to illustrate these concepts.

1. Introduction

For over two decades, real-time optimization has evolved to standard practice in the chemical and petroleum industry. The ability to optimize predictive models provides a major step towards linking on-line performance to higher-level corporate planning decisions. As described in Grossmann (2005); Kadam and Marquardt (2007); Engell (2007), these decisions form a hierarchy as seen in Fig. 1, with levels of decision-making that include planning, scheduling, site-wide and real-time optimization, model predictive control and regulatory control. In this pyramid, note that the frequency of decision-making increases from top to bottom, while the overall impact of decision-making increases from bottom to top. Moreover, optimization models for decision-making have been developed for all but the bottom-most level. Here, planning and scheduling decision models are often characterized by linear models with many discrete decisions. These are usually represented as mixed integer linear programs (MILPs), and occasionally mixed integer nonlinear programs (MINLPs) that capture key nonlinear elements. On the other hand, site-wide and real-time optimization require nonlinear process models which usually reflect steady-state performance of the plant, while model predictive control (MPC) is often characterized by linear dynamic models.

Communication and interaction among levels requires decisions made at higher levels to be feasible at lower levels. Moreover, the performance described by lower level models must be reflected accurately in decisions made at higher levels. Clearly, the strongest communication and interaction is through direct integration of optimization formulations between two or more levels. Such integration has been described for planning and scheduling (Grossmann, 2005), as well as with dynamic optimization for batch processes (Bhatia & Biegler, 1996; Flores-Tlacuahuac & Grossmann, 2006). Similarly, site-wide and real-time levels can be integrated through compatible steady-state optimization models. However, as noted and analyzed extensively in Yip and Marlin (2004), integrated MPC and real-time optimization may suffer inconsistencies due to mismatch of linear dynamic and steady state nonlinear models, and also because of conflicting objectives. While this inconsistency can be tolerated for processes with decoupled dynamics (e.g., in refineries with fast rejection of disturbances and slow dynamic trends), it remains severe for nonlinear processes, especially those that are never in steady state. Examples of these include batch processes, power plants and

* Corresponding author. Tel.: +1 412 268 2232; fax: +1 412 268 7139.
E-mail addresses: lb01@andrew.cmu.edu (L.T. Biegler), vzavala@andrew.cmu.edu (V.M. Zavala).

0098-1354/$ – see front matter © 2008 Elsevier Ltd. All rights reserved.
doi:10.1016/j.compchemeng.2008.08.006

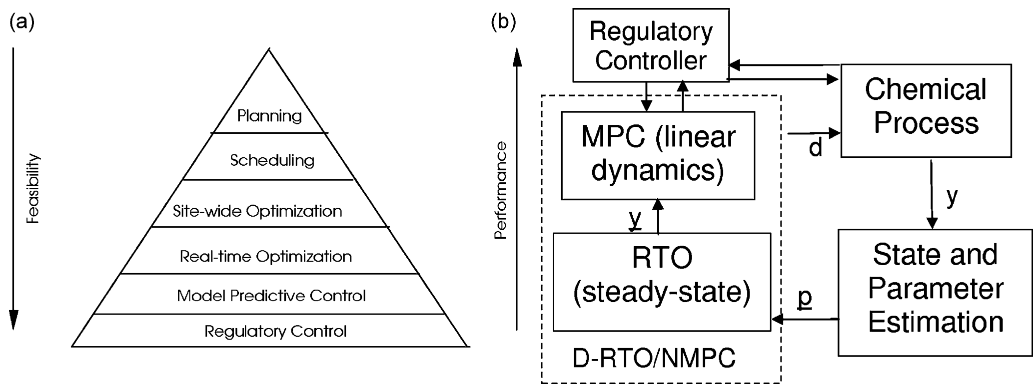

Fig. 1. (a) Decision-making hierarchy in operations. (b) Structure for on-line optimization.

polymerization processes with load changes and grade transitions, and production units that operate in a periodic manner, such as simulated moving beds (SMBs) (Toumi, Diehl, Engell, & Schlöder, 2005) and pressure swing adsorption (PSA) (Jiang, Biegler, & Fox, 2003).

As motivated in Fig. 1b, treating these nonlinear processes requires on-line optimization with nonlinear dynamic models, including strategies such as nonlinear model predictive control (NMPC) (Bartusiak, 2007). Research in this direction includes development and application of detailed and accurate first-principles differential-algebraic equation (DAE) models for off-line dynamic optimization. Numerous case studies (Busch, Oldenburg, Santos, Cruse, & Marquardt, 2007) have demonstrated the effectiveness of off-line dynamic optimization. In addition, a comprehensive research effort on real-time dynamic optimization is described in Grötschel, Krumke, and Rambau (2001) and several large-scale industrial NMPC applications have been reported (Nagy, Franke, Mahn, and Allgöwer, 2007). Moreover, in addition to enabling NLP solvers, there is a better understanding of NMPC stability properties and associated dynamic optimization problem formulations that provide them (Camacho & Bordons, 2007; Magni & Scattolini, 2007). Finally, from recent activity in dynamic real-time optimization, it is clear that with improved optimization formulations and algorithms, the role of NMPC can be greatly expanded to include economic objectives and multiple operating stages over the predictive horizon (with transitions due to product change-overs, nonstandard cyclic operations, or anticipated shutdowns) (Engell, 2007; Grötschel et al., 2001).

In this study we explore optimization formulations that merge RTO and MPC tasks and replace them with NMPC strategies that perform the role of dynamic real-time optimization, as shown in Fig. 1. This approach stems from recent work based on recently developed state estimation and control strategies that rely on NLP sensitivity for on-line calculations (Zavala, Laird, & Biegler, 2008). As a result, on-line computational costs remain negligible even for very large optimization models. In the next section we briefly describe the dynamic optimization problem and summarize the simultaneous collocation approach used to solve it. Section 3 then discusses on-line strategies for NMPC and moving horizon estimation (MHE) based on NLP sensitivity, for dynamic optimization and state estimation, respectively. This is followed in Section 4 by a brief presentation of a recent case study on grade transitions for a realistic LDPE process. Finally, Section 5 summarizes these concepts and outlines areas for future work.

2. Large-scale dynamic optimization

Consider the optimization problem with N dynamic stages in the following form:

$$\min \sum_{k=1}^{N} \varphi(z(t_k), u_k) \tag{1a}$$

$$\text{s.t.} \quad \frac{\mathrm{d}z_k(t)}{\mathrm{d}t} = f(z_k(t), w_k(t), u_k), \quad t \in [t_{k-1}, t_k] \tag{1b}$$

$$g(z_k(t), w_k(t), u_k) = 0 \tag{1c}$$

$$z_k(t_{k-1}) = z_{k-1}(t_{k-1}) \tag{1d}$$

$$u_k^{\mathrm{L}} \le u_k(t) \le u_k^{\mathrm{U}} \tag{1e}$$

$$w_k^{\mathrm{L}} \le w_k(t) \le w_k^{\mathrm{U}} \tag{1f}$$

$$z_k^{\mathrm{L}} \le z_k(t) \le z_k^{\mathrm{U}}, \quad k = 1, \ldots, N \tag{1g}$$

where $z_k(t) \in \Re^{n_z}$ is the vector of state variables in stage k, $u_k \in \Re^{n_u}$ is the vector of manipulated variables, and $w_k(t) \in \Re^{n_w}$ is a vector of algebraic variables. As constraints we have the differential and algebraic equations (DAEs) (1b) and (1c) which we assume without loss of generality are index one.

A number of approaches can be taken to solve problem (1). These include *sequential methods*, also known as *control vector parameterization*, *multiple shooting* and the *simultaneous collocation approach*. In this last approach, we discretize both the state and control profiles in time using collocation on finite elements to form a large NLP. Equivalent to a fully implicit Runge-Kutta method with high order accuracy and excellent stability properties, this discretization is also a desirable way to obtain accurate solutions for boundary value problems and related optimal control problems. With this simultaneous approach, the DAE system is solved only once, at the optimal point, and therefore difficult intermediate solutions are avoided. Also, control variables are discretized at the same level as the state variables, so that under mild conditions (see Hager, 2000; Kameswaram & Biegler, 2008) the Karush Kuhn Tucker (KKT) conditions of the NLP are consistent with the optimality conditions of the discretized variational problem; fast convergence rates to variational solutions have also been shown. Finally, the simultaneous collocation approach deals with unstable systems in a straightforward manner and allows direct enforcement of state and control variable constraints, at the same level of discretization as the state variables of the DAE system.

On the other hand, simultaneous collocation approaches do require efficient, large-scale optimization strategies (Biegler, Cervantes, & Wächter, 2002; Jockenhövel, Biegler, & Wächter, 2003) because they directly couple the solution of the DAE system with the optimization problem. To address the resulting large-scale NLP, a full space, interior point (or barrier) solver, called IPOPT (Wächter & Biegler, 2006), has recently been developed, which solves large-scale NLPs very efficiently. IPOPT applies a logarithmic barrier method to inequality constraints in the NLP, solves a set of equality constrained optimization problems for a monotonically decreasing sequence of the barrier parameter, and quickly converges to the solution of the original NLP. This sequence of problems is solved with sparse Newton method applied to the KKT conditions. The IPOPT algorithm enjoys excellent convergence properties and exhibits superior performance, especially on large dynamic optimization problems.

As a result of the optimization formulation and solution algorithm, the simultaneous collocation approach has much lower complexity bounds than competing dynamic optimization strategies. As shown in Zavala, Laird, and Biegler (2008), a particular advantage of the simultaneous approach is that exact second derivatives can be obtained very cheaply and the expensive DAE integration and sensitivity steps are avoided. Also, note that as the number of discretized control profiles increases, one has significant computational advantages with the simultaneous collocation approach. Moreover, IPOPT can be adapted to different problem structures and can accommodate a wide variety of linear decomposition methods. This allows the efficient solution of very large NLPs on the order of several *million* variables, constraints and degrees of freedom (Hagemann, Schenk, & Wächter, 2005; Zavala, Laird, & Biegler, 2008c).

3. On-line dynamic optimization

As seen in Fig. 2, realization of on-line dynamic optimization requires two elements. First, the current state of the process and the model parameters must be estimated from the process measurements (the moving horizon estimator (MHE) considered here). Next, with the updated model, optimal values of the manipulated variables need to be calculated (performed here by the NMPC controller). These elements are time-critical applications that require fast and reliable algorithms, as the updated manipulated variables require optimization of two problems represented by (1).

However, efficient NLP formulations and fast NLP solvers are not enough for large-scale on-line dynamic optimization. Any *computational delay* means that the applied control is based on state information that no longer represents the current plant. Moreover, large delays can degrade performance and even destabilize the process (Findeisen & Allgöwer, 2004). Therefore, to incorporate first principle models for on-line dynamic optimization, it is essential

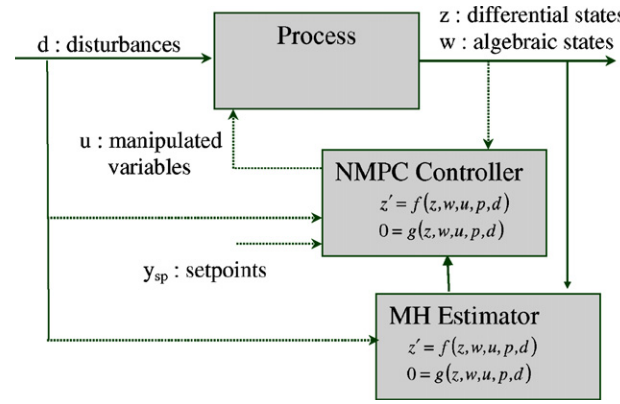

Fig. 2. Tasks for dynamic on-line optimization.

to separate these complex optimization computations into *background* and *on-line* components and to minimize the cost of the on-line component. Here we assume that solution of both the MHE and NMPC problems can be done within one or more sampling intervals in "background" for an initial condition "close" to the measured (or estimated) state. Once the new process state is estimated, a perturbed problem is then solved to update the NLP solution.

3.1. NMPC formulation

To describe this approach, we first consider the optimization problem for nonlinear model predictive control written over the moving time horizon shown in Fig. 3. After temporal discretization with collocation on finite elements, the dynamic optimization problem (1) for $x(k)$, the state available at time t_k, can be written as

$$\begin{aligned} \mathcal{P}_N(x(k)) \quad & \min_{v_l} J(x(k)) = \Psi(z_N) + \sum_{l=0}^{N-1} \psi(z_l, v_l) \\ \text{s.t.} \quad & z_{l+1} = f(z_l, v_l), \quad l = 0, \ldots, N-1, \quad z_0 = x(k), \\ & z_l \in \mathbb{X}, \quad z_N \in \mathbb{X}_f, \quad v_l \in \mathbb{U} \end{aligned} \tag{2}$$

For this optimization problem we require the following assumptions so that $J(x(k))$ is a Lyapunov function and we can guarantee stability of the closed-loop system.

Assumption 1 *(Nominal stability assumptions of NMPC).*

- The terminal penalty $\Psi(\cdot)$, satisfies $\Psi(z) > 0$, $\forall z \in \mathbb{X}_f \setminus \{0\}$ (Magni & Scattolini, 2007).

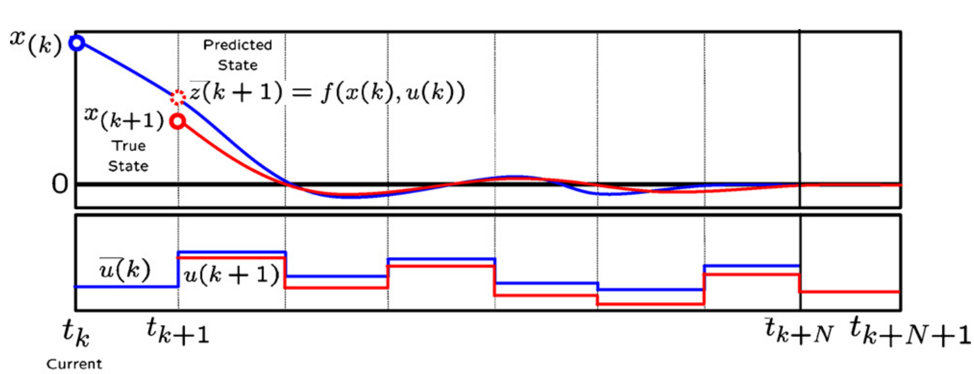

Fig. 3. NMPC moving horizon problem.

- There exists a local control law $u = h_f(z)$ defined on $\mathbb{X}_f$, such that $f(z, h_f(z)) \in \mathbb{X}_f, \forall z \in \mathbb{X}_f$, and $\Psi(f(z, h_f(z))) - \Psi(z) \leq -\psi(z, h_f(z)), \forall z \in \mathbb{X}_f$.
- The optimal stage cost $\psi(x, u)$ satisfies $\alpha_p(|x|) \leq \psi(x, u) \leq \alpha_q(|x|)$ where $\alpha_p(\cdot)$ and $\alpha_q(\cdot)$ are $\mathcal{K}$ functions, i.e., $\alpha(0) = 0, \alpha(s) > 0, \forall s > 0$ and strictly increasing.

It should be noted that while not all economic functions apply, the above property of the ψ functions is still sufficiently general to allow economic terms and thus serve as a Lyapunov function. It should be noted that the use of economic objectives within this MPC formulation is also addressed in a number of studies (Bartusiak, 2007; Engell, 2007; Odloak, Zanin, & Tvrzska de Gouvea, 2002; Sbarbaro & Johansen).

From the solution of this problem, we obtain $u(k) = v_0$ and inject it into the plant. In the nominal case, this drives the state of the plant towards $x(k+1) = z(k+1) = f(x(k), u(k))$. Once $x(k+1)$ is known, the prediction horizon is shifted forward by one sampling instant and problem $\mathcal{P}_N(x(k+1))$ is solved to find $u(k+1)$. This recursive strategy gives rise to the feedback law, $u(k) = h^{\mathrm{id}}(x(k))$ which we call the ideal NMPC controller (neglecting computational delay).

Now consider the state of the plant, $x(k)$, at t_k and that we already have the control action $u(k)$. In the *nominal case* the system evolves according to the dynamic model in (8) starting at t_k, and we can predict the future state exactly (i.e., $x(k+1) = z(k+1) = f(x(k), u(k))$) and compute the future control action by solving $\mathcal{P}_N(z(k+1))$ *in advance*. If this problem can be solved before t_{k+1}, then $u(k+1) = h^{\mathrm{id}}(x(k+1))$ will already be available *without on-line computational delay*. Moreover, it is easy to prove (Zavala & Biegler, in press) that this strategy has identical nominal stability properties as the standard or ideal NMPC controller.

On the other hand, a realistic controller must also be robust to model mismatch, unmeasured disturbances and measurement noise. Here, ideal NMPC provides a mechanism to react to these features with some inherent robustness. In particular, tolerance to mismatch and disturbances can be characterized by input-to-state stability (Magni & Scattolini, 2007; Zavala & Biegler, in press). The key to a realistic extension of our NMPC strategy is to note that problem $\mathcal{P}_N(z(k+1))$ is parametric in its initial conditions, with parameters given by $p_0 = z(k+1)$ and $p = x(k+1)$. One can represent $\mathcal{P}_N(z(k+1))$ as

$$\min F(\mathbf{x}, p_0) \quad \text{s.t.} \quad c(\mathbf{x}, p_0) = 0, \quad \mathbf{x} \geq 0 \tag{3}$$

where $\mathbf{x}$ contains all the variables of problem (2). We note that the interior-point solver IPOPT solves (3) by applying Newton's method to the following equations:

$$\nabla F(\mathbf{x}, p_0) + \nabla c(\mathbf{x}, p_0)\lambda - \nu = 0 \tag{4a}$$

$$c(\mathbf{x}, p_0) = 0 \tag{4b}$$

$$XV = \mu e \tag{4c}$$

where $X = \mathrm{diag}(\mathbf{x})$, $V = \mathrm{diag}(\nu)$, and the barrier parameter μ is gradually reduced to zero so that the solution sequence of (4) converges to the solution of (3). From the optimality conditions of (3) evaluated at the solution $\mathbf{x}_*$ one can obtain, under mild regularity conditions of the NLP (Fiacco, 1983), a second order estimate of the perturbed solution to

$$\min F(\mathbf{x}, p), \quad \text{s.t.} \quad c(\mathbf{x}, p) = 0, \quad \mathbf{x} \geq 0 \tag{5}$$

i.e., $\Delta\mathbf{x} \approx \mathbf{x}_*(p) - \mathbf{x}_*(p_0)$, from the linear system:

$$\begin{bmatrix} W_* & A_* & -I \\ A_*^{\mathrm{T}} & 0 & 0 \\ V_* & 0 & X_* \end{bmatrix} \begin{bmatrix} \Delta\mathbf{x} \\ \Delta\lambda \\ \Delta\nu \end{bmatrix} = -\begin{bmatrix} \nabla L(\mathbf{x}_*, \lambda_*, \nu_*, p) \\ c(\mathbf{x}_*, p) \\ 0 \end{bmatrix} \tag{6}$$

where W_* is the Hessian of the Lagrangian function $L(\mathbf{x}, \lambda, \nu) = F(\mathbf{x}) + c(\mathbf{x}, p_0)^{\mathrm{T}}\lambda - \mathbf{x}^{\mathrm{T}}\nu$, and $A_* = \nabla c(\mathbf{x}_*, p_0)$. Because the KKT matrix in (14) is identical to the iteration matrix used in IPOPT, it is already available in factorized form. Hence, once the next state is known, $\Delta\mathbf{x}$ is formed and the desired approximate solution can be obtained with a single on-line backsolve. As described in Zavala et al. (2008a) and Zavala and Biegler (in press), this on-line step usually requires less than 1% of the dynamic optimization calculation.

The resulting advanced step NMPC (as-NMPC) controller consists of the following steps:

1. Having $x(k)$ and $u(k)$, obtain evaluate $z(k+1) = f(x(k), u(k))$ and solve $\mathcal{P}_N(z(k+1))$ in background.
2. Once the measured (or estimated) state $x(k+1)$ is obtained from the plant, obtain $u(k+1)$ on-line as a perturbed solution using the linear system (14) derived from $\mathcal{P}_N(z(k+1))$ with $p_0 = z(k+1)$ and $p = x(k+1)$.
3. Set $k = k+1$ and return to Step 1.

Note from the above steps that the advanced step NMPC controller is able to handle the nonlinearity of the system since it updates the KKT matrix at each time step, while avoiding the difficulty of computational delay. Moreover, as expressed by the following property, as-NMPC has input-to-state robustness properties that are essentially as strong as ideal NMPC.

Theorem 1 (Robust Stability of advanced step NMPC). Under Assumption 1, the cost function $J(x)$ obtained from the advanced-step controller is an input-to-state Lyapunov function and the resulting as-NMPC controller is input-to-state stable (Zavala & Biegler, in press).

3.2. Moving horizon estimation

Moving horizon estimation (MHE) has emerged as a superior alternative for state estimation over traditional observers and Kalman filters. As shown recently in Rao, Rawlings, and Mayne (2003), MHE has desirable asymptotic stability properties and compares very well to extended Kalman filters (EKF) (Haseltine & Rawlings, 2005). MHE requires the on-line solution of dynamic optimization problems, but sensitivity-based optimization (as in the previous section) can also be applied to these problems (Zavala et al., 2008b). Here we apply similar concepts based on background and on-line calculations to the NLP for the k th horizon of an MHE problem. From problem (1) the MHE problem can be represented by

$$\begin{aligned} \mathcal{M}_N(\eta(k)) \quad & \min_{z_0, \xi_l} \phi(\eta(k)) \\ &= (z_0 - \bar{z}_0(k))^{\mathrm{T}} \mathbf{\Pi}_0{}^{-1}(k)(z_0 - \bar{z}_0(k)) + (y_k - h(z_N))^{\mathrm{T}} \mathbf{R}_N{}^{-1}(y_k - h(z_N)) \\ &\quad + \sum_{l=0}^{N-1} (y(k-N+l) - h(z_l))^{\mathrm{T}} \mathbf{R}_l{}^{-1}(y(k-N+l) - h(z_l)) \\ &\quad + \sum_{l=0}^{N-1} \xi_l^{\mathrm{T}} \mathbf{Q}_l{}^{-1} \xi_l \quad \text{s.t.} \quad z_{l+1} = f(z_l, u_l) + \xi_l, \quad l = 0, \ldots, N-1, \\ & z_l \in \mathbb{Z}, \quad \xi_l \in \Xi \end{aligned} \tag{7}$$

where $y(k) \in \Re^{n_y}$ is the vector of measurements, $h(z)$ is the output prediction based on the state, $\xi_l \in \Re^{n_\xi}$ is a vector of process noise variables, $\bar{z}_0(k)$ is the current prior reference for the initial state with covariance $\mathbf{\Pi}_0{}^{-1}(k)$, and $\eta(k)$ are the problem data containing the current measurement history. The objective function has a

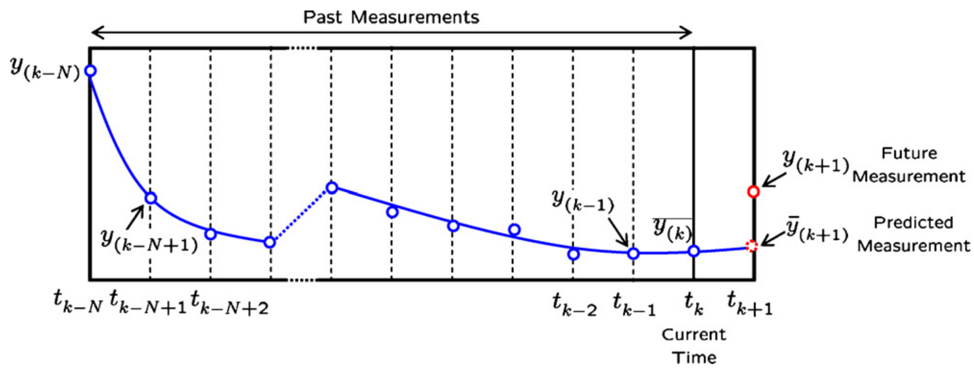

Fig. 4. Moving horizon formulation with actual measurements (closed circles) and predicted measurements (dotted circles).

typical least-squares form that includes the arrival cost, deviations for measured variables and minimization of process noise. This formulation is the dual of the NMPC problem with a DAE model in the same form as $\mathcal{P}_N(x)$. To partition the background and on-line calculations, at time t_k and having the current state estimate $x(k)$ and control $u(k)$ we predict the future measurement through forward simulation $z(k+1) = f(x(k), u(k))$, $\bar{y}(k+1) = h(z(k+1))$ as shown by the open circles in Fig. 4.

The development of the as-MHE strategy follows along the same lines as in the previous section. The predicted problem is solved in background using IPOPT and the sensitivity system (14) is constructed. Once new measurement $y(k+1)$ is received at t_{k+1}, we replace the predicted measurement $\bar{y}(k+1)$ by its measured counterpart; these are treated as parameters p_0 and p, respectively. Using the algorithm described in Zavala et al. (2008b), nonlinear models can be updated and estimated on-line with very little on-line cost, and fast MHE and NMPC can be done on-line with large-scale first principle models. As with as-NMPC, nonlinear MHE problems can be solved with on-line calculations that usually require less than 1% of the time to solve the dynamic optimization problem.

4. LDPE grade transition case study

To demonstrate the advantages of the as-NMPC and as-MHE strategies, we consider operating scenarios for a high-pressure low-density polyethylene (LDPE) process described in Cervantes, Tonelli, Brandolin, Bandoni, and Biegler (2002) and Zavala et al. (2008a). As seen in Fig. 5, ethylene is polymerized in a long tubu-

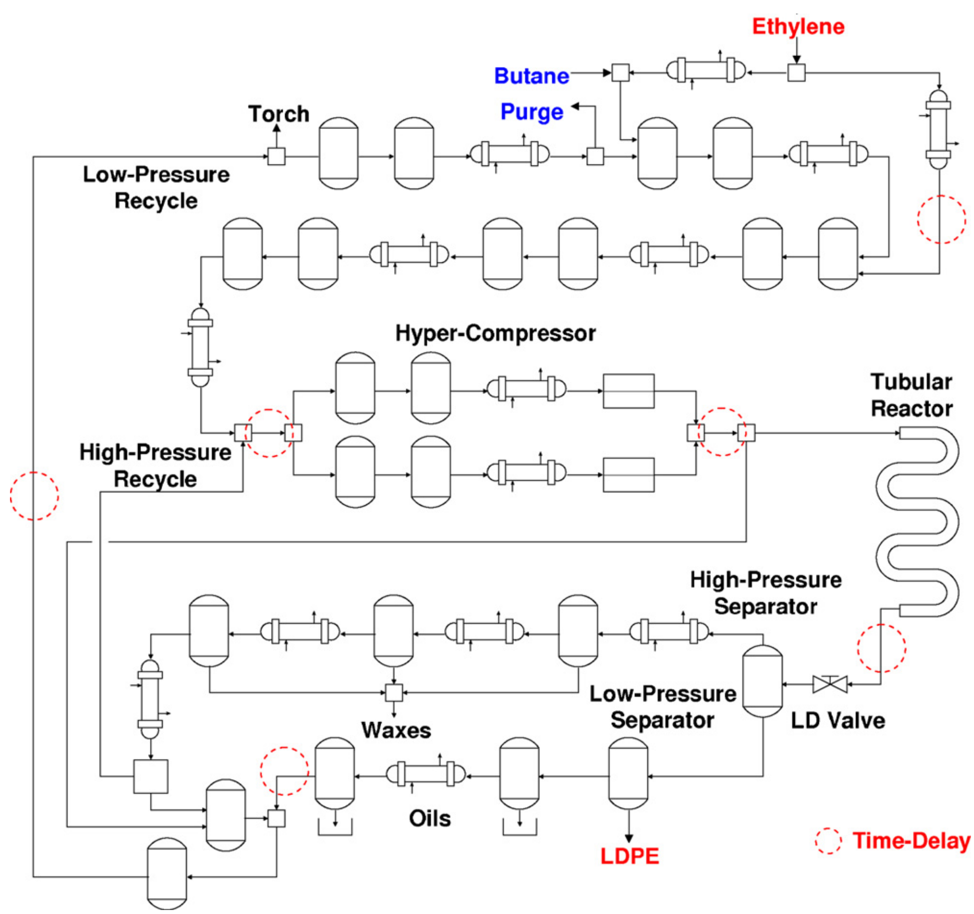

Fig. 5. High-pressure LDPE process flowsheet.

lar reactor at high pressures (2000–3000 atm) and temperatures (150–300°C) through a free-radical mechanism. Accordingly, a large number of compression stages is required to obtain these extreme operating conditions. The final product is recovered by flash separation. These flexible processes obtain several different product grades by adjusting the reactor operating conditions. Desired properties such as polymer melt index are obtained by control of the reactor temperature, pressure and concentration of a chain-transfer agent (usually butane and propylene).

The process represents a difficult dynamic system; reactor dynamics are much faster than responses in the recycle loops and long time delays are present throughout the compression and separation systems. Due to the complex, exothermic nature of the polymerization, the reactor temperature and pressure are enforced strictly along the operating horizon following fixed recipes. The main operational problem in these processes consists of providing fast adjustments to the butane feed and purge stream to keep the melt index at a desired reference value. This is done during different operating stages such as grade transitions (switching between two different operating points) and normal operation (disturbance rejection). The resulting DAE model of this process contains over 350 state variables and differential-algebraic equations. A detailed description of the process model can be found in Cervantes et al. (2002).

4.1. Performance of NMPC controller

We now consider an appropriate optimal feedback policy that minimizes the switching time between steady states corresponding to the production of different polymer grades. This poses a severe test of the as-NMPC algorithm as it needs to optimize over a large dynamic transition. The NMPC problem solved on-line at every sampling time t_k is given by

$$\min \int_{t_k}^{t_{k+N}} (w_{C_4}(t) - w_{C_4}^{r})^2 + (F_{C_4}(t) - F_{C_4}^{r})^2 + (F_{pu}(t) - F_{pu}^{r})^2 \, dt$$
$$\text{s.t.} \quad \text{DAEs for LDPE model with } z(t_k) = x(k) \tag{8}$$

where the inputs are the flowrates of butane and purge streams, F_{C_4} and F_{pu}, respectively, the output is the butane weight fraction in the recycle stream, w_{C_4}, and superscript 'r' denotes a reference value. Here we consider equal control and prediction horizons. Note that while the objective in this study is of the tracking type, it has a direct economic effect in that it minimizes the transition time to a new polymer grade. Moreover, in a more recent study (Zavala & Biegler, 2008), we have also considered the LDPE production rate as the objective function within this NMPC framework.

Using the simultaneous collocation approach, problem (16) is converted into a large-scale NLP with 15 finite elements with 3 collocation points in each element. The resulting NLP contains 27,135 constraints, and 30 degrees of freedom. For the dynamic optimization, long prediction times on the order of hours are used with sampling times on the order of minutes; here we set $N = 15$ and $\Delta t = t_{k+1} - t_k$=6 min. Note that the *background* NLP must be solved in under this time.

In this case study, we ignore the computational delay that affects the closed-loop response in order to assess the best behavior of the ideal NMPC controller. Also, the plant response is obtained by introducing both strong and random disturbances to the time delays in the recycle loops. The performance of the NMPC approaches is presented in Fig. 6. Note that the optimal feedback policy involves the saturation of both control valves for the first 2500 s of operation, with the final flow rates set to values corresponding to the new operating point. It is interesting to observe that the output profile for as-NMPC is indistinguishable from the full optimal solution, with only small differences in the input profiles.

The on-line and background computational times are also worth comparing. Ideal NMPC requires around 351 CPU seconds and about 10 IPOPT iterations of on-line computation while as-NMPC requires only 1 CPU second by using only a single backsolve to obtain the updated solution vector. In addition to reducing on-line computation by over 300 times, as-NMPC also serves as an excellent basis for effective initialization of the *next* NLP problem solved in background. From the perturbed solution provided by the sensitivity calculation (14), as-NMPC provides very accurate initializations for all the sampling times. Leading to only 2–3 IPOPT iterations, as-NMPC also reduces the *background* NLP computation by up to a factor of 5.

4.2. Performance of fast moving horizon estimator

To demonstrate the MHE algorithm on the LDPE process, we choose the butane concentration in the recycle loop as the output measurement. Measurement data are generated by simulation of the dynamic model, using fixed control profiles over a horizon with 60 sampling points. This output profile is then corrupted using

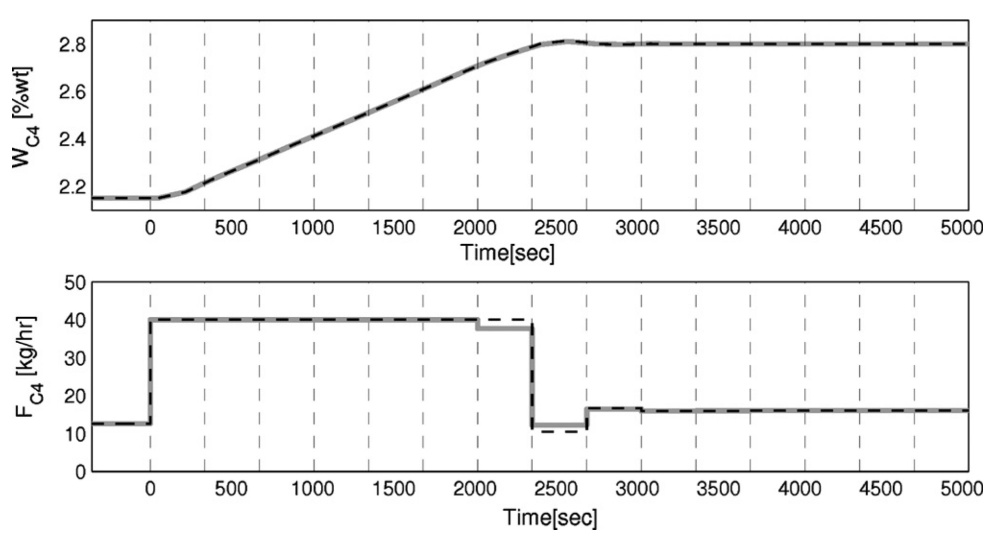

Fig. 6. Closed-loop performance of the ideal NMPC (solid) and as-NMPC (dashed) approaches with output w_{C_4} and inputs F_{C_4} and F_{pu}.

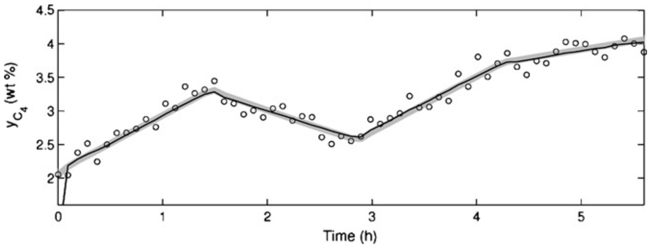

Fig. 7. Measured, estimated and true profiles of the output variable for LDPE case study.

Gaussian noise with $\sigma = 0.05$. The resulting least-squares objective function:

$$\min_{\mathbf{z}_0} (\mathbf{z}_0 - \bar{\mathbf{z}}_0)^{\mathrm{T}} \Pi_0^{-1} (\mathbf{z}_0 - \bar{\mathbf{z}}_0) + \sum_{l=0}^{N} \frac{1}{\sigma^2} (y_{C_4}(k - N + l) - \bar{y}_{C_4}(l))^2 \quad (9)$$

and the model equations are used for the formulation of the estimation problem. Here, $y_{C_4}(l), \bar{y}_{C_4}(l)$ are the butane measurement and prediction at sampling time t_k, respectively, vector $\mathbf{z}_0 \in \Re^{294}$ contains the initial conditions for all the states with a given *a priori* estimate $\bar{z}_0$ obtained from simulation and $\Pi_0^{-1} = 10\,\mathbb{I}_{n_z}$. With $N = 15$ finite elements and 3 collocation points for the dynamic model, the resulting NLP contains 27,121 constraints and 295 degrees of freedom.

Computational results are reported for this problem in Zavala et al. (2008b). Here IPOPT requires an average of 6 iterations to solve the MHE problem and about 200 CPU seconds. The majority of this CPU time is devoted for the factorization of the KKT matrix. Consequently, a standard MHE algorithm would introduce an on-line computational delay of more than 3 min. On the other hand, the as-MHE algorithm needs only a single backsolve on-line, which requires less than a single CPU second. Moreover, the approximate solutions obtained from NLP sensitivity can be used to warm-start the algorithm for the solution of the background nominal problems at each sampling time, thus reducing the cost of the background problems to only 3–5 iterations.

Finally, as shown in Zavala et al. (2008b) a byproduct of the IPOPT optimization allows us to determine easily that this solution of the MHE problem yields observable state estimates with the given measurement data. Fig. 7 presents the measured, estimated and true profiles of the output variable along 60 sampling times. Here the fast MHE algorithm is able to estimate accurately the true output variable, and the noise perturbations do not induce drastic changes between neighboring problems.

5. Conclusions

Integration of optimal decision-making for operations is an important task for today's highly competitive chemical industry. In particular, tying higher-level decisions to optimal on-line operations is essential. This paper considers challenges posed by real-time optimization and model predictive control interactions, as they both comprise large-scale NLP problems, often with conflicting models and objectives. We show that simultaneous collocation and fast NLP solvers, like IPOPT, allow the integration of real-time optimization and model predictive control through a single dynamic optimization formulation. In addition, concepts from NLP sensitivity allow on-line calculations for state estimation, control and dynamic optimization to be drastically reduced with almost all of the optimization calculations performed in a background step. These concepts are demonstrated on a large LDPE process where on-line optimization computations were reduced by over two orders of magnitude, with negligible loss of performance.

The capability to solve large optimization problems with negligible on-line costs leads to a number of opportunities for dynamic real-time optimization. In particular, extensions of MHE and NMPC problems can be formulated to include on-line economic optimization over much longer time horizons. This becomes especially important for multi-stage process decisions which require optimal transitions between different operating modes. These challenging large-scale NLPs can be handled by specialized decomposition strategies that can be exploited by IPOPT. Finally, the on-line realization of dynamic optimization requires treatment of model mismatch and parametric uncertainty, noise models that capture process and measurement errors, and capabilities for fault detection and optimal recovery from upsets. All of these need to be considered within the dynamic optimization strategy and remain as topics for future work.

References

Bartusiak, R. D. (2007). NLMPC: A platform for optimal control of feed- or product-flexible manufacturing. In *Assessment and future directions of NMPC* (p. 338). Berlin: Springer.

Betts, J. (2001). *Practical methods for optimal control using nonlinear programming*. Philadelphia, PA, USA: SIAM Series on Advances in Design and Control.

Bhatia, T., & Biegler, L. T. (1996). Dynamic optimization in the design and scheduling of multiproduct batch plants. *Industrial & Engineering Chemical Research*, *35*(7), 2234.

Biegler, L. T., Cervantes, A. M., & Wächter, A. (2002). Advances in simultaneous strategies for dynamic process optimization. *Chemical Engineering Science*, *57*, 575–593.

Busch, J. J., Oldenburg, J., Santos, M., Cruse, A., & Marquardt, W. (2007). Dynamic predictive scheduling of operational strategies for continuous processes using mixed-logic dynamic optimization. *Computers and Chemical Engineering*, *31*, 574–587.

Camacho, E. F., & Bordons, C. (2007). Nonlinear model predictive control: An introductory survey. In *Assessment and future directions of NMPC* (pp. 1–16). Berlin: Springer.

Cervantes, A. M., Tonelli, S., Brandolin, A., Bandoni, J. A., & Biegler, L. T. (2002). Large-scale dynamic optimization for grade transitions in a low density polyethylene plant. *Computers & Chemical Engineering*, *26*, 227.

Engell, S. (2007). Feedback control for optimal process operation. *Journal of Process Control*, *17*, 203–219.

Fiacco, A. V. (1983). *Introduction to sensitivity and stability analysis in nonlinear programming*. New York: Academic Press.

Findeisen, R., & Allgöwer, F. (2004). Computational delay in nonlinear model predictive control. In *Proceedings of the International Symposium on Advanced Control of Chemical Processes, ADCHEM'03*.

Flores-Tlacuahuac, A., & Grossmann, I. E. (2006). Simultaneous cyclic scheduling and control of a multiproduct CSTR. *Industrial & Engineering Chemical Research*, *45*(20), 6698–6712.

Franke, R., & Doppelhamer, J. (2007). Integration of advanced model based control with industrial it. In *Assessment and future directions of NMPC* (p. 368). Berlin: Springer.

Grossmann, I. E. (2005). Enterprise-wide optimization: A new frontier in process systems engineering. *AIChE Journal*, *51*(7), 1846.

Grötschel, S., Krumke, J., & Rambau (Eds.). (2001). *Online optimization of large systems*. Berlin: Springer.

Hagemann, M., Schenk, O., & Wächter, A. (2005). *Combinatorial approaches to the solution of saddlepoint problems in large-scale parallel interior-point optimization*. Research Report RC 23824. IBM T.J. Watson Research Center, Yorktown, USA, December.

Hager, W. W. (2000). Runge-Kutta methods in optimal control and the transformed adjoint system. *Numerical Mathematics*, *87*, 247–282.

Haseltine, E. L., & Rawlings, J. B. (2005). Critical evaluation of extended Kalman filtering and moving horizon estimation. *Industrial & Engineering Chemical Research*, *44*, 8.

Jiang, L., Biegler, L. T., & Fox, V. G. (2003). Simulation and optimization of pressure swing adsorption systems for air separation. *AIChE Journal*, *49*(5), 1140–1157.

Jockenhövel, T., Biegler, L. T., & Wächter, A. (2003). Dynamic optimization of the Tennessee Eastman process using the optcontrolcentre. *Computers and Chemical Engineering*, *27*(11), 1513–1531.

Kadam, J., & Marquardt, W. (2007). Integration of economical optimization and control for intentionally transient process operation. In *Assessment and future directions of NMPC* (p. 419). Berlin: Springer.

Kameswaram, S., & Biegler, L. T. (2008). Convergence rates for direct transcription of optimal control problems using collocation at Radau points. *Computational Optimization and Applications*, *41*(1), 81–126.

Kawajiri, Y., & Biegler, L. T. (2006). Optimization strategies for simulated moving bed and powerfeed processes. *AIChE Journal*, *52*(4), 1343–1350.
Magni, L., & Scattolini, R. (2007). Robustness and robust design of MPC for nonlinear systems. In *Assessment and future directions of NMPC*. Berlin: Springer.
Nagy, Z. K., Franke, R., Mahn, B., & Allgöwer, F. (2007). Real-time implementation of nonlinear model predictive control of batch processes in an industrial framework. In *Assessment and future directions of NMPC* (p. 429). Berlin: Springer.
Odloak, D., Zanin, A. C., & Tvrzska de Gouvea, M. (2002). Integrating real-time optimization into the model predictive controller of the fcc system. *Control Engineering Practice*, *10*, 819–831.
Qin, S., & Badgwell, T. (2000). An overview of nonlinear model predictive control applications. In F. Allgöwer & A. Zheng (Eds.), *Nonlinear model predictive control*. Basel: Birkhaeuser.
Rao, C. V., Rawlings, J. B., & Mayne, D. Q. (2003). Constrained state estimation for nonlinear discrete time systems: Stability and moving horizon approximations. *IEEE TAC*, *48*, 2.
Sbarbaro, D., & Johansen, T. A. (2005). Lyapunov-based optimizing control of nonlinear blending processes. *IEEE Transactions on Control Systems Technology*, *13*(4), 631–638.
Toumi, A., Diehl, M., Engell, S., Bock, H. G., & Schlöder, J. P. (2005). Finite horizon optimizing control of advanced SMB chromatographic processes. In *16th IFAC World Congress*.
Wächter, A., & Biegler, L. T. (2006). On the implementation of a primal-dual interior point filter line search algorithm for large-scale nonlinear programming. *Mathematical Programming*, *106*(1), 25–57.
Ydstie, B. E., Perea, E., & Grossmann, I. E. (2003). A model predictive control strategy for supply chain optimization. *Computers & Chemical Engineering*, *27*, 1201–1218.
Yip, W. S., & Marlin, T. (2004). The effect of model fidelity on real-time optimization performance. *Computers and Chemical Engineering*, *28*, 267.
Zavala, V. M., & Biegler, L. T. (2008). Large-scale nonlinear programming strategies for the operation of low-density polyethylene tubular reactors. In *Proceedings of ESCAPE-18* (pp. 629–634).
Zavala, V., & Biegler, L. T. (in press). The advanced-step NMPC controller: Optimality, stability and robustness. *Automatica* (http://dynopt.cheme.cmu.edu/papers/papers.html).
Zavala, V. M., Laird, C. D., & Biegler, L. T. (2008). Fast implementations and rigorous models: Can both be accommodated in NMPC? *International journal of Robust and Nonlinear Control*, *18*, 800–815.
Zavala, V. M., Laird, C. D., & Biegler, L. T. (2008b). A fast moving horizon estimation algorithm based on nonlinear programming sensitivity. *Journal of Process Control, 18*(9), 876–884.
Zavala, V. M., Laird, C. D., & Biegler, L. T. (2008c) Interior-point decomposition approaches for parallel solution of large-scale nonlinear parameter estimation problems. *Chemical Engineering Science, 63*(19), 4834–4845.