Efficient Nonlinear Programming Algorithms for Chemical Process Control and Operations

Lorenz T. Biegler

Abstract Optimization is applied in numerous areas of chemical engineering including the development of process models from experimental data, design of process flowsheets and equipment, planning and scheduling of chemical process operations, and the analysis of chemical processes under uncertainty and adverse conditions. These *off-line* tasks require the solution of nonlinear programs (NLPs) with detailed, large-scale process models. Recently, these tasks have been complemented by *time-critical, on-line* optimization problems with differential-algebraic equation (DAE) process models that describe process behavior over a wide range of operating conditions, and must be solved sufficiently quickly. This paper describes recent advances in this area especially with dynamic models. We outline large-scale NLP formulations and algorithms as well as NLP sensitivity for on-line applications, and illustrate these advances on a commercial-scale low density polyethylene (LDPE) process.

1 Introduction

Manufacturing processes for petroleum products, basic chemicals, pharmaceuticals, specialty chemicals, consumer products, agricultural chemicals and fertilizers form essential and irreplaceable components of our day-to-day existence. In the US alone, these products lead to revenues of over $\$10^{12}$/yr. Their manufacture is dominated by raw material and energy costs and a strong competitive market, which drives down operating margins. These factors emphasize the need for systematic, model-based process optimization strategies, both in the original design of the process and in day-to-day operations.

Lorenz T. Biegler
Chemical Engineering Department, Carnegie Mellon University, Pittsburgh, PA 15213 USA,
e-mail: biegler@cmu.edu

Mathematical models for process optimization reflect processing tasks such as mixing, reaction and separation at appropriate conditions, through calculation of state variables, e.g., stream flowrates, temperature, pressure and composition. Modeling equations include conservation laws for mass, energy, and momentum along with constitutive relations and equilibrium conditions (such as physical properties, hydraulics, rate laws and interface behavior). Moreover, with advances in computing hardware and numerical algorithms there has been a steady evolution of model sophistication from steady state to dynamic behavior and from lumped to spatially distributed systems.

Nonlinear programming strategies have been used for process optimization for almost 50 years. These have been essential for plant and equipment design, retrofitting and operations planning. Over the past 25 years *real-time optimization* has also evolved as a standard practice in the chemical and petroleum industry. In particular, the ability to optimize predictive models provides a major step towards linking on-line performance to higher-level corporate planning decisions. As described in [11, 15], these tasks form a well-known pyramidal hierarchy with levels of decision-making including planning at the top, followed by scheduling, site-wide and real-time optimization, model predictive control and regulatory control at the bottom. In this pyramid, the frequency of decision-making increases from top to bottom, while the impact and importance of decision-making increases from bottom to top. Moreover, while planning and scheduling decision models are often characterized by linear models with many discrete decisions, site-wide and real-time optimization require detailed nonlinear process models which usually reflect steady-state performance of the plant. On the other hand, model predictive control (MPC) is often formulated with linear dynamic models.

Interaction among decision-making levels requires that higher-level actions be feasible at lower levels. Moreover, the performance described by lower level models must be reflected accurately in decisions made at higher levels. A particularly close integration is needed for real-time optimization and control, especially for nonlinear processes that may never really be in steady state. Examples of these include batch processes, processes with load changes and grade transitions, such as power plants and polymerization processes, and production units that operate in a periodic manner, such as Simulated Moving Beds (SMBs) [17] and Pressure Swing Adsorption (PSA) [14]. Treating these nonlinear processes requires on-line optimization with nonlinear dynamic models, including strategies such as nonlinear model predictive control (NMPC) [2]. Research in this direction includes development and application of detailed and accurate first-principle differential-algebraic equation (DAE) models for off-line dynamic optimization [5, 15, 23]. A comprehensive research effort on real-time dynamic optimization is described in [12] and, more recently, large-scale industrial NMPC applications have been reported at ExxonMobil [2], BASF [21] and ABB [10]. Moreover, in addition to enabling NLP solvers, there is a much better understanding of NMPC stability properties and associated dynamic optimization problem formulations that provide them (see [20]). Along with these theoretical developments, NMPC robustness properties have also been developed and analyzed [19]. From the comprehensive treatment of dynamic real-time optimization in [12], it is clear that with improved optimization formulations and algorithms, the role of on-line dynamic optimization can be greatly expanded to consider economic objectives directly, allow longer time horizons with additional constraints and degrees of freedom to improve the objective, and incorporate multiple operating stages over the predictive horizon, including transitions in the predictive horizon due to product change-overs, nonstandard cyclic operations, or anticipated shutdowns [24, 12].

The next section provides a background of dynamic optimization strategies and their application to process optimization. An LDPE (low density polyethylene) process case study is also introduced to illustrate the application of these strategies. Sect. 3 then considers simultaneous collocation methods for *off-line* dynamic optimization. A parameter estimation for the LDPE reactor is presented to demonstrate the effectiveness of this approach. Sect. 4 then discusses methods for dynamic optimization for *on-line, time-critical* applications and introduces an NLP sensitivity-based nonlinear model predictive controller, which relies on a "background solution" of the NLP optimization. This is illustrated on a grade transition optimization for the LDPE process. Finally, Sect. 5 concludes the paper and outlines areas for future work.

2 Background

To develop the NLP formulation and solution strategy we consider the dynamic optimization problem in the following form:

$$\min \quad \varphi(z(t_f)) \tag{1}$$

$$\text{s.t.} \quad \frac{dz(t)}{dt} = f(z(t), y(t), u(t), p), \;\; z(0) = z_0 \tag{2}$$

$$g(z(t), y(t), u(t), p) = 0, \;\; t \in [0, t_f] \tag{3}$$

$$g_f(z(t_f)) = 0 \tag{4}$$

$$u_L \le u(t) \le u_U, \; y_L \le y(t) \le y_U, \; z_L \le z(t) \le z_U \tag{5}$$

where $t \in [t_0, t_f]$ (e.g., time) is the independent variable, $z(t) \in \Re^{n_z}$ is the vector of differential state variables, $u(t) \in \Re^{n_u}$ is the vector of control variables, $y(t) \in \Re^{n_y}$ is a vector of algebraic state variables, and p is a set of optimization variables independent of time. The process model is described by semi-explicit differential and algebraic equations (DAEs) (2),(3) which we assume without loss of generality, are index one.

A number of approaches can be taken to solve (1)-(5). Until the 1970s, these problems were solved using an *indirect* or *variational approach*, based on the first order necessary conditions for optimality obtained from Pontryagin's Maximum Principle; a review of these approaches can be found in [12]. However, if the problem requires the handling of active inequality constraints, finding the correct switching structure as well as suitable initial guesses for state and adjoint variables may be difficult. This limitation has made the *indirect* approach less popular for NMPC applications and can be overcome by *direct* methods that apply NLP solvers.

Sequential methods with NLP solvers, also known as *control vector parameterization*, represent the control variables as piecewise polynomials [26] and perform the optimization with respect to the coefficients of these polynomials. Given initial conditions and a set of control parameters, the DAE model is solved over time within an inner loop of the NLP iteration; the control variables are then updated by the NLP solver itself. Gradients of the objective function with respect to the control coefficients and parameters are calculated either from direct DAE sensitivity equations or by integration of the adjoint equations. Sequential strategies are relatively easy to construct and to apply as they contain the components of reliable DAE solvers (e.g., DASSL, DASOLV, DAEPACK) and NLP solvers (e.g., NPSOL, SNOPT). On the other hand, repeated numerical integration of the DAE model is required, which may become time consuming for large problems. Moreover, sequential approaches may fail with unstable dynamics [1]. Instead, for unstable systems *Multiple Shooting*, which inherits many of the advantages of sequential approaches should be applied. Here, the time domain is partitioned into N time elements, i. e., $t \in [t_{k-1}, t_k], k = 1, \dots N$, and the DAE models are integrated separately in each element [4, 8]. Control variables are parameterized as in the sequential approach and gradient information is obtained for both control variables as well as the initial conditions of the state variables in each element. Finally, equality constraints are added in the NLP to link the elements and ensure that the states are continuous across each element. As with the sequential approach, bound constraints for states and controls are normally imposed only at the grid points t_k.

In the *simultaneous collocation approach*, also known as *direct transcription*, we represent both the state and control profiles as piecewise polynomials in time using collocation on finite elements $t \in [t_{k-1}, t_k], k = 1, \dots N$. This approach corresponds to a fully implicit Runge-Kutta method with high order accuracy and excellent stability properties. It is also a desirable way to obtain accurate solutions for boundary value problems and related optimal control problems. On the other hand, simultaneous approaches also require efficient, large-scale optimization strategies [7, 3] because they directly couple the solution of the DAE system with the optimization problem. The DAE system is solved only once, at the optimal point, and therefore can avoid intermediate solutions that require excessive computational effort or may not even exist. Moreover, in the *simultaneous approach* the control variables can be discretized at the same level as the state variables and, under mild conditions, (see [13, 16]) the Karush-Kuhn-Tucker (KKT) conditions of the simultaneous NLP are consistent with the optimality conditions of the discretized variational problem, and fast convergence rates to the solution of the variational problem have been shown. Moreover, simultaneous approaches can deal with unstable systems and allow the direct enforcement of state and control variable constraints, at the same level of discretization as the state variables of the DAE system.

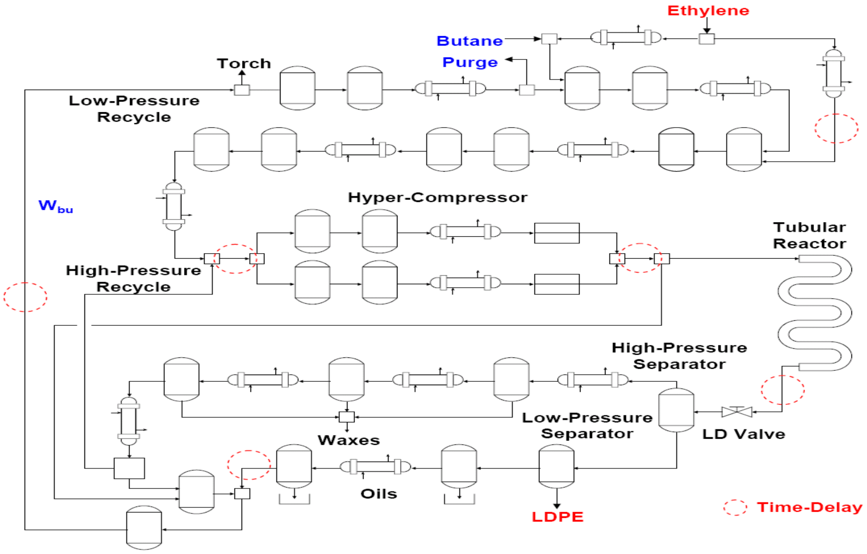

Fig. 1 High-pressure LDPE Process Flowsheet

2.1 LDPE Case Study

Low density polyethylene (LDPE) is currently the most widely produced polymer. Its uses span many packaging applications including plastic bags, food wrap, squeeze bottles, and plastic films in construction; different polymer grades are produced to ensure the best material properties for each of these applications. The high-pressure process for LDPE manufacture is described in [6, 30] and serves as a case study for dynamic optimization. As seen in Fig. 1, ethylene is polymerized in a long tubular reactor at high pressures (1500-3000 atm) and temperatures (130-300 ^{o}C) through a free-radical mechanism. Accordingly, many compression stages are required to obtain these extreme operating conditions. The LDPE product is recovered after several stages of vapor-liquid separation. These flexible processes obtain several different polymer grades by adjusting the reactor operating conditions. The process model contains a number of challenges for optimization. In the next section we will focus on parameter estimation of a detailed LDPE reactor model, while in Sect. 4, we deal with the important on-line problem of grade changes.

3 Simultaneous Collocation Approach

The DAE optimization problem can be converted into an NLP by approximating state and control profiles by piecewise polynomials on finite elements ($t_0 < t_1 < \ldots < t_N = t_f$). Using a monomial basis representation for the differential profiles, which is popular for Runge-Kutta discretizations, leads to:

$$z(t) = z_{i-1} + h_i \sum_{q=1}^{K} \Omega_q \left(\frac{t - t_{i-1}}{h_i} \right) \frac{dz}{dt}_{i,q} \tag{6}$$

where z_{i-1} is the value of the differential variable at the beginning of element i, h_i is the length of element i, $dz/dt_{i,q}$ is the value of its first derivative in element i at the collocation point q, and Ω_q is a polynomial of order K, satisfying

$$\Omega_q(0) = 0,\ \Omega'_q(\rho_r) = \delta_{q,r},\ \ q,r = 1,\ldots,K, \tag{7}$$

where $\rho_r \in [0,1]$ is the normalized location of the r-th collocation point within each element. Continuity of the differential profiles is enforced by

$$z_i = z_{i-1} + h_i \sum_{q=1}^{K} \Omega_q\,(1)\, \frac{dz}{dt}_{i,q}. \tag{8}$$

From a number of studies (see [1, 16]), we prefer Radau collocation points (with $\rho_K = 1$) as it has a stronger stability property. In addition, the control and algebraic profiles are approximated using a Lagrange basis representation of the form:

$$y(t) = \sum_{q=1}^{K} \psi_q \left(\frac{t - t_{i-1}}{h_i} \right) y_{i,q}, \quad u(t) = \sum_{q=1}^{K} \psi_q \left(\frac{t - t_{i-1}}{h_i} \right) u_{i,q}, \tag{9}$$

where $y_{i,q}$ and $u_{i,q}$ represent the values of the algebraic and control variables, respectively, in element i at collocation point q. ψ_q is the Lagrange polynomial of degree $K-1$ satisfying $\psi_q(\rho_r) = \delta_{q,r}$ for $q,r = 1,\ldots,K$. From (6), the differential variables are required to be continuous throughout the time horizon, while the control and algebraic variables are allowed to have discontinuities at the boundaries of the elements. Substitution of (9) into (1)-(5) leads to the following NLP:

$$\min_{\frac{dz}{dt}_{i,q}, u_{i,q}, y_{i,q}, p} \quad \varphi(z_N) \tag{10}$$

$$\begin{gathered}
\text{s.t.} \qquad \frac{dz}{dt}_{i,q} = f(z_{i,q}, y_{i,q}, u_{i,q}, p),\ \ g(z_{i,q}, y_{i,q}, u_{i,q}, p) = 0 \\
u_{i,q} \in [u_L, u_U],\ y_{i,q} \in [y_L, y_U],\ z_{i,q} \in [z_L, z_U], \quad i = 1,\ldots N,\ q = 1,\ldots K \\
\text{and } \ (6),\ (8),\ g_f(z_N) = 0
\end{gathered}$$

This NLP can be rewritten as:

$$\min_{\mathbf{x} \in \Re^n} \varphi(\mathbf{x}),\ \ \text{s.t. } c(\mathbf{x}) = 0,\ \mathbf{x}_L \le \mathbf{x} \le \mathbf{x}_U \tag{11}$$

where $\mathbf{x} = \left(\frac{dz}{dt}_{i,q}, z_i, y_{i,q}, u_{i,q}, p \right)^T$, $f : \Re^n \longrightarrow \Re$ and $c : \Re^n \longrightarrow \Re^m$. To address the resulting large-scale NLP, we apply a full space, interior point (or barrier) solver, embodied in a code called IPOPT. IPOPT applies a Newton strategy to the optimality conditions that result from the primal-dual barrier subproblem,

$$\min \ \varphi(\mathbf{x}) - \mu \sum_{i=1}^{n} [\ln(\mathbf{x}^{(i)} - \mathbf{x}_L^{(i)}) + \ln(\mathbf{x}_U^{(i)} - \mathbf{x}^{(i)})], \ \text{s.t. } c(\mathbf{x}) = 0. \tag{12}$$

Problem (12) is solved for a sequence of decreasing values of the barrier parameter μ; under typical regularity conditions this sequence of solutions $\mathbf{x}(\mu)$ converges to the solution of (11) [9].

The IPOPT code [27] includes a novel filter based line-search strategy and also allows the use of exact second derivatives. Under mild assumptions, the filter-based barrier algorithm has global and superlinear convergence properties; correspondingly the IPOPT code performs very well when compared to state-of-the-art NLP solvers. Originally developed in FORTRAN, the IPOPT code was recently redesigned to allow for structure dependent specialization of all linear algebra operations. Implemented in C++ and freely available through the COIN-OR foundation, IPOPT can be obtained from the following website:
`http://projects.coin-or.org/Ipopt`.

A key step in the IPOPT algorithm is the solution of linear systems derived from the linearization of the first order optimality conditions (in primal-dual form) of the barrier subproblem. The linear KKT system can be solved with any direct linear solver configured with IPOPT. However, as the problem size grows, the time and memory requirements can make this approach expensive. Instead, specialized decompositions such as Schur complements lead to efficient (and often parallizable) solution strategies. This allows the efficient solution of very large NLPs on the order of several *million* variables, constraints and degrees of freedom [22]. A detailed description of IPOPT's internal decomposition features and their implementation in the IPOPT software environment is given in [18, 25].

Because of these features, the simultaneous collocation approach has lower complexity bounds than competing dynamic optimization strategies, especially since exact second derivatives can be obtained very cheaply and expensive DAE integration and direct sensitivity steps are avoided. This comparison and complexity analysis can be found in [31].

3.1 Parameter Estimation for LDPE Reactor

An important off-line optimization problem is the estimation of reactor parameters from experimental data. The LDPE tubular reactor seen in Fig. 1 can be described as a jacketed, multi-zone device with a predefined sequence of reaction and cooling zones. Different configurations of monomer and initiator mixtures enter in feed and multiple sidestreams, and are selected to maximize the reactor productivity and obtain desired polymer properties. The total reactor length ranges between 0.5 to 2 km, while its internal diameter does not exceed 70-80 mm. Models of this reactor typically comprise detailed polymerization kinetic mechanisms and reacting mixture thermodynamic and transport properties at extreme conditions. A first-principles model describing the gas-phase free-radical homopolymerization of ethylene in the presence of several different initiators and chain-transfer agents at supercritical conditions is considered in [30]. The reaction mechanism consists of 35 reactions with 100 kinetic parameters for each polymer chain of a given length. Here, the method of moments is used to describe macromolecular properties of the copolymer including number- and weight-average molecular weights and polydispersity as described in [25, 30]. The steady-state evolution of the reacting mixture along the multiple reactor zones can be formulated as a multi-stage DAE system of the form,

$$\begin{aligned} &\mathbf{F}_{k,j}\left[\frac{dz_{k,j}(t)}{dt}, z_{k,j}(t), y_{k,j}(t), p_{k,j}, \mathit{\Pi}\right] = 0 \\ &z_{k,j}(0) = \phi(z_{k,j-1}(t_{L_{k,j-1}}), w_{k,j-1}),\ \ k = 1,\ldots,NS,\ j = 1,\ldots,NZ \end{aligned} \tag{13}$$

where the stage index j denotes a particular reactor zone and index k pertains to a product grade or operating scenario; this formulation allows estimation over different reactor configurations. At zone boundaries, these DAE models are coupled through material and energy balances $\phi(\cdot)$ while additional inputs, $w_{k,j}$, are introduced for monomer, initiator, and cooling water. Also, $t_{L_{k,j}}$ denotes the total length of zone j in scenario k, $p_{k,j}$ denotes local parameters (such as heat transfer coefficients and initiator efficiencies) in each zone j and scenario k and $\mathit{\Pi}$ corresponds to the kinetic rate constants which apply to all stages. The reactor model contains around 130 ordinary differential equations and 500 algebraic equations for each instance k. Because of significant coupling among the state variables and parametric sensitivity, the reactor DAE model is also highly nonlinear and stiff.

Using (13), we estimate kinetic parameters, $\mathit{\Pi}$, to match the plant reactor operating conditions and polymer properties. However, due to the uncertainty associated to the fouling and initiator decomposition mechanisms, it is also necessary to include the local parameters as well. To capture the interaction of $p_{k,j}$ and $\mathit{\Pi}$ and to account for the measurement errors in the multiple of flow rates, concentrations, temperatures and pressures around the reactor, we consider a multi-scenario estimation problem of the form:

$$\begin{aligned} &\min_{\mathit{\Pi}, p_{k,j}, w_{k,j}} \sum_{k=1}^{NS}\sum_{j=1}^{NZ_k}\sum_{i=1}^{NM_{k,j}} \left(y_{k,j}(t_i) - \bar{y}_{k,j,i}\right)^T \mathbf{V}_{\mathbf{y}}^{-\mathbf{1}} \left(y_{k,j}(t_i) - \bar{y}_{k,j,i}\right) \\ &\quad + \sum_{k=1}^{NS}\sum_{j=1}^{NZ_k} \left(w_{k,j} - w_{k,j}^M\right)^T \mathbf{V}_{\mathbf{w}}^{-\mathbf{1}} \left(w_{k,j} - w_{k,j}^M\right) \\ &\text{s.t. (13), } \mathbf{H}_{k,j}\left[z_{k,j}(t), y_{k,j}(t), p_{k,j}, \mathit{\Pi}\right] \le 0 \end{aligned} \tag{14}$$

where the output variables are matched to the corresponding available plant measurements for each operating scenario or data set k. The vector of outputs contains the reactor temperature profile, jacket inlet and outlet temperatures in each zone, as well as macromolecular properties and product quality at the reactor outlet.

To formulate the estimation problem (14) as a multi-scenario NLP, we perform a full discretization of the differential and algebraic variables and group the resulting set of variables by data sets or scenarios k. For each data set, we use a total of 16 finite elements for the reaction zones, 2 finite elements for the cooling zones and 3 collocation points for the discretization in (14), so that each scenario has around 12,000 constraints and 92 degrees of freedom (corresponding to 32 local parameters $p_{k,j}$, 25 global parameters Π and 35 input variables $w_{k,j}$). In order to obtain exact first and second derivative information, the NLP instances are implemented as NS separate AMPL models that internally *indicate* the set of variables corresponding to the global parameters Π.

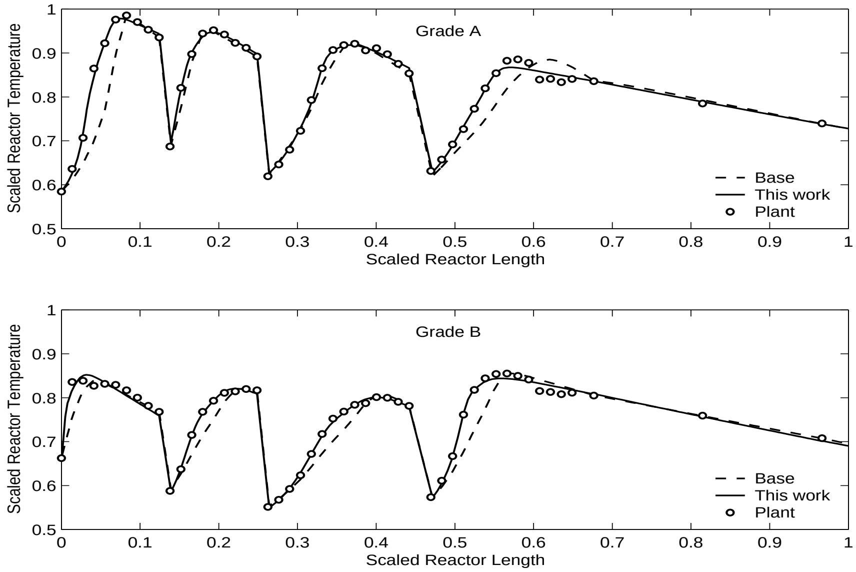

Fig. 2 Comparison of reactor temperature profiles for using simultaneous and "zone-by-zone" parameter estimation.

Using the internal decomposition strategy in optimization strategy we consider the solution of multi-scenario NLPs with $NS \leq 32$ data sets. A result of the model fit to two typical product grade data sets can be seen in Fig. 2, where the dashed line depicts a suboptimal "zone-by-zone" estimation with global parameters fixed. The results were obtained in a Beowulf type cluster using standard Intel Pentium IV Xeon 2.4GHz, 2GB RAM processors running under Linux. These are compared against serial solutions of the multi-scenario problems on a single processor with similar characteristics. Fig. 3 presents both computational results. The serial solution of the multi-scenario NLPs exhausts the available memory when the number of data sets exceeds nine, while the parallel implementation overcomes this memory bottleneck and solves problems with up to 32 data sets. For the parallel approach, notice that the effect of parallelism is reflected less in the time required per iteration than in the time per factorization of the KKT matrix. Nevertheless, we see that the time per iteration can be consistently kept below 5 seconds, while the factorization in the serial approach can take as much as 35 seconds before running out of memory. More information on this application and details of the optimization strategy can be found in [25, 30].

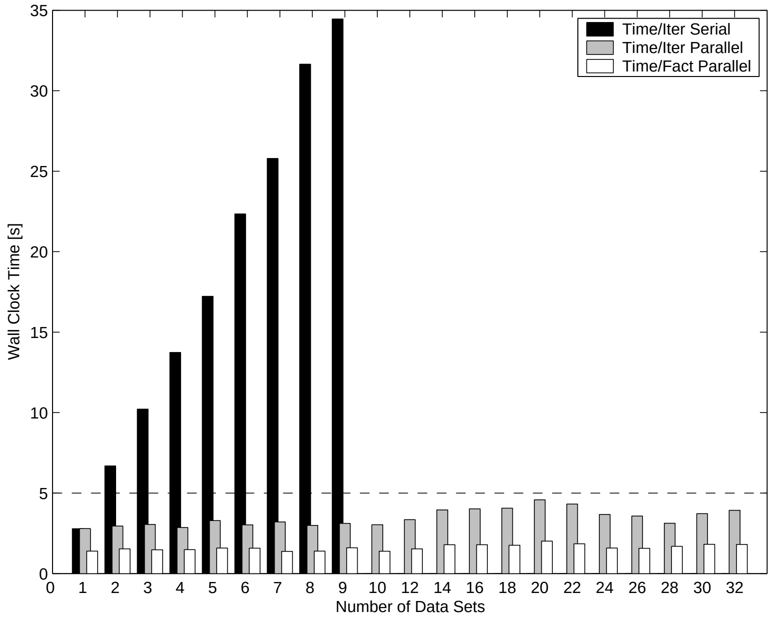

Fig. 3 Wall clock time per iteration and per KKT matrix factorization for multi-scenario parameter estimation with IPOPT. Serial and parallel implementations.

4 Fast NMPC based on IPOPT Sensitivity

As described in the previous section, efficient dynamic optimization solvers enable fast solution times even for large-scale models. However, on-line optimization demands time-limited, robust calculations that may exceed the capabilities of current solvers. To address this issue, we now explore the concept of sensitivity-based real-time dynamic optimization with rigorous, first principle process models. To address these concepts, consider the moving time horizon shown in Fig. 4, with sampling times $t_{k+l}, l = 0, \ldots N$. For chemical processes we note that sampling intervals are usually on the order of minutes. On the other hand, once the current plant state $x(k)$ is known, the appropriate control action $u(k)$ must be available to the plant. Any *computational delay* in determining $u(k)$ will lead to a deterioration of performance and even destabilization of the on-line optimization.

To satisfy these restrictions, we partition the optimization calculations into *background* and *on-line* steps. We assume the NLP can be solved within only a few sampling intervals in "background" for an initial condition "close" to the measured (or estimated) state. Once this state is obtained, a perturbed problem is solved quickly to update the NLP solution, using a particular NLP sensitivity formulation.

To describe this approach, we consider the dynamic optimization problem written over the moving time horizon shown in Fig. 4. After temporal discretization, the dynamic optimization problem can be written as the following simplified NLP,

$$\mathscr{P}_N(x(k)): \quad \min_{z_{l|k}, v_{l|k}} \Phi(z_{N|k}) + \sum_{l=0}^{N-1} \psi(z_{l|k}, v_{l|k}) \tag{15}$$

$$\begin{aligned} \text{s. t. } & z_{l+1|k} = f(z_{l|k}, v_{l|k}),\ l = 0, \ldots N-1 \\ & z_{0|k} = x(k),\ z_{l|k} \in \mathbb{X},\ z_{N|k} \in \mathbb{X}_f,\ v_{l|k} \in \mathbb{U}, \end{aligned} \tag{16}$$

where $z_{l|k}$ and $v_{l|k}$ are the states and controls, respectively, over the prediction horizon. From the solution of this problem for current time t_k, we obtain $u(k) = v_{0|k}$ and inject it into the plant. In the nominal case, this drives the state of the plant towards $x(k+1) = z_{1|k} = f(x(k), u(k))$. Once $x(k+1)$ is known, the prediction horizon is shifted forward by one sampling interval and problem $\mathscr{P}_N(x(k+1))$ is solved to find $u(k+1)$. This recursive strategy gives rise to the ideal NMPC controller (neglecting computational delay).

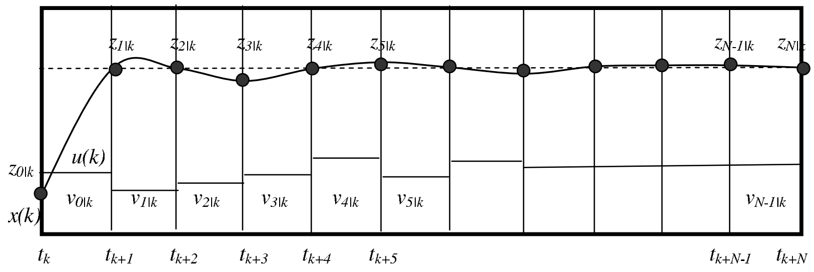

Fig. 4 NMPC moving horizon problem

Now consider the state of the plant at the previous sampling time, $x(k-1)$, where we already have the control $u(k-1)$. In the nominal case the system evolves according to the dynamic model (16), and we can predict the future state by solving $\mathscr{P}_N(f(x(k-1), u(k-1)))$ in advance. For instance, if this problem can be solved between t_{k-1} and t_k, then $u(k)$ will already be available at t_k. For this, we define the equivalent NLP of the form,

$$
\begin{aligned}
\mathscr{P}_{N+1}(x(k-1),u(k-1)):\ \min_{z_{l|k-1},v_{l|k-1}}\ & \Phi(z_{N|k-1})+\psi(x(k-1),u(k-1)) \\
& +\sum_{l=0}^{N-1}\psi(z_{l|k-1},v_{l|k-1}) \\
\text{s.t. }\ & z_{l+1|k-1}=f(z_{l|k-1},v_{l|k-1}),\ l=0,\dots N-1 \\
& z_{0|k-1}=f(x(k-1),u(k-1)), \\
& z_{l|k-1}\in\mathbb{X},\ z_{N|k-1}\in\mathbb{X}_f,\ v_{l|k-1}\in\mathbb{U}
\end{aligned}
\tag{17}
$$

In the nominal case, it is clear that the solution of this problem is *equivalent to* $\mathscr{P}_N(x(k))$ and that $\mathscr{P}_{N+1}(x(k-1),u(k-1))$ can be solved in advance to obtain $u(k)=v^*_{0|k-1}$ *without computational delay*. Moreover, under the NMPC assumptions posed in [20], it is easy to see that such a controller has the *same nominal stability properties* as the ideal NMPC controller [28].

On the other hand, a realistic controller must also be robust to model mismatch, unmeasured disturbances and measurement noise. As noted in [19], ideal NMPC provides a mechanism to react to these features along with some inherent robustness. In particular, tolerance to mismatch and disturbances can be characterized by input-to-state stability [19, 28]. In [28] we focus on sensitivity-based NMPC schemes and show their inherent robustness properties through input-to-state stability concepts. The key to this extension comes by noting that problem $\mathscr{P}_{N+1}(x(k),u(k))$ is parametric in its initial conditions so we can define the dummy parameter vector $p_0=x(k)$. Here we rewrite $\mathscr{P}_{N+1}(x(k),u(k))$ as the following NLP,

$$
\min\varphi(\mathbf{x},p_0),\ \text{ s.t. }\ c(\mathbf{x},p_0)=0,\ \mathbf{x}_L\le\mathbf{x}\le\mathbf{x}_U \tag{18}
$$

and we define $\mathbf{x}$ as the vector of all variables in $\mathscr{P}_{N+1}(x(k),u(k))$. From the optimality conditions of (18), and under mild regularity conditions of the NLP [9], we obtain a first order estimate of the perturbed solution of (18), i.e., $\Delta\mathbf{x}=\mathbf{x}_*(p)-\mathbf{x}_*(p_0)$. This can be calculated very cheaply in IPOPT from the factorization of the KKT matrix in the final NLP iteration. Therefore in the presence of uncertainty, we apply the sensitivity equations of $\mathscr{P}_{N+1}(x(k),u(k))$ to find the approximate solution of $\mathscr{P}_N(x(k+1))$.

Moreover, to maintain a consistent active set for the solution of $\mathscr{P}_N(x(k+1))$, we modify the sensitivity calculation to determine the value of p that enforces the relation $z_{0|k}=x(k+1)$ in the perturbed $\mathscr{P}_{N+1}(x(k),u(k))$, instead of a direct change ($\Delta p=x(k+1)-x(k)$) in the initial conditions. Coupled to the linearized optimality conditions, the added constraint, $\Delta z_{0|k}=x(k+1)-z^*_{0|k}$, gives rise to an extended set of linear sensitivity equations, which can be solved efficiently through a Schur complement approach. This approach takes advantage of the already factorized KKT matrix at the solution of $\mathscr{P}_{N+1}(x(k),u(k))$ with IPOPT. Therefore, once the next state is known, the desired approximate solution can be obtained from the background Schur decomposition and a single on-line backsolve [31]. As described in [31, 28], the on-line step requires less than 1% of the (already fast) dynamic optimization calculation. We denote this sensitivity-based approach the Advanced Step

NMPC (as-NMPC) controller. This controller can be viewed as a fast linear model predictive controller *linearized about the optimal nonlinear model at the previous measurement*. Moreover, it inherits the stability and robustness properties of ideal NMPC while avoiding the difficulties of computational delay.

4.1 NMPC for LDPE Process

To demonstrate the advantages of the sensitivity-based NMPC strategy, we return to the LDPE case study. 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 especially important during transitions (switching between two different operating points). As shown in [6], dynamic optimization can lead to significant reduction in the grade transition time; in one case, it was reduced from about 5 h to no more than 2.8 h, leading to reduction of at least 23 tons of off-spec product.

The resulting DAE model of the LDPE process (with a simplified reactor model) contains 289 differential and 64 algebraic state variables. 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 NMPC algorithm as it needs to optimize over a large dynamic transition. The following moving horizon problem is solved on-line at every sampling time t_k:

$$\begin{aligned} \min \int_{t_k}^{t_{k+N}} \left(w_{C_4}(t) - w^r_{C_4}\right)^2 + \left(F_{C_4}(t) - F^r_{C_4}\right)^2 + \left(F_{Pu}(t) - F^r_{Pu}\right)^2 dt \\ \text{s.t. DAEs for LDPE Model} \end{aligned} \tag{19}$$

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. Using the simultaneous collocation approach, problem (19) 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, we set $N = 15$ and sampling interval to 6 min.

To compare ideal and as-NMPC strategies, we ignore the effect of computational delay in the closed-loop response. To assess robust performance, the plant response is also subjected to strong, random disturbances in the transportation delays in the recycle loops. Performance of both NMPC approaches is presented in Fig. 5. Note that the optimal feedback policy involves the saturation of both control valves for the first 2500 seconds of operation, with the final flowrates 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.

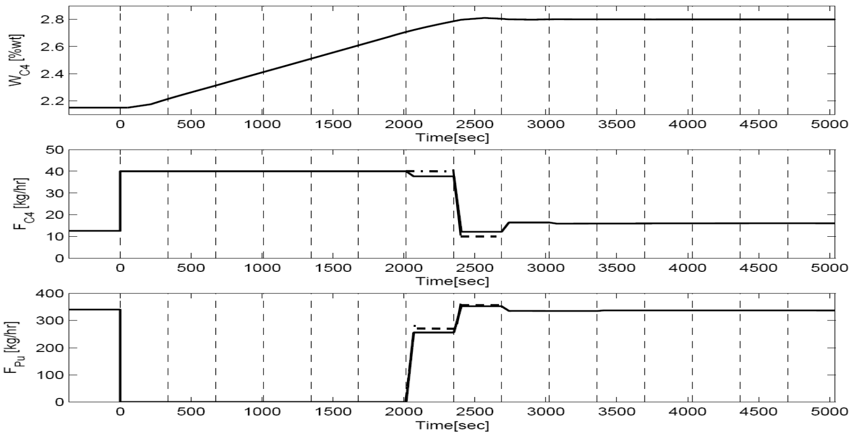

Fig. 5 Closed-loop performance of the ideal NMPC (solid) and as-NMPC (dashed) approaches with output w_{C_4} and inputs F_{C_4} and F_{Pu}

The on-line and background computational times are especially worth comparing. Ideal NMPC requires around 351 CPU seconds and about 10 IPOPT iterations of on-line computation while as-NMPC requires a negligible on-line time (1.04 CPU seconds) for the solution of the Schur complement system and a final backsolve to obtain the updated solution vector. As a result, as-NMPC reduces the on-line computation time (and associated computational delay) by over two orders of magnitude with virtually no loss in performance. Moreover, 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, as-NMPC provides very accurate NLP initializations at all sampling times. Leading to only 2-3 IPOPT iterations, as-NMPC also reduces the *background* NLP computation by up to a factor of five.

5 Conclusions

This paper addresses the increasing value of dynamic optimization for chemical process operations. Both off-line and on-line optimization tasks demand fast and robust optimization strategies, often for challenging large-scale applications. Current dynamic optimization formulations and algorithms are reviewed with an emphasis on the simultaneous collocation approach. This strategy has advantages for unstable systems, and with the suitable application of large-scale NLP solvers (such as IPOPT), it is especially effective for time-critical applications. Moreover, for on-line applications, NLP sensitivity can be calculated very cheaply from IPOPT; this leads to a nonlinear model predictive control strategy with fast on-line performance and minimal computational delay. All of these aspects are demonstrated on a case study for a large-scale polymerization process.

Nevertheless, this summary represents only a beginning in addressing dynamic real-time optimization. Future challenges include effective off-line solution strategies for large, multi-stage dynamic optimization problems along with a tighter integration of planning and scheduling decisions. On-line strategies can also benefit from moving horizon estimation (MHE) which incorporates nonlinear dynamic models. A sensitivity-based MHE strategy was developed recently and exhibited very fast performance as well as accurate state estimates [29]. In addition, more robust on-line dynamic optimization problem formulations are needed to include model uncertainty and disturbance models. Finally, further significant impacts can be made through dynamic optimization on challenging large-scale process applications, such as the LDPE process.

Acknowledgements Advice and assistance from several research colleagues including Victor Zavala, Carl Laird, Andreas Wächter and Shiva Kameswaran is gratefully acknowledged. The author is also pleased to aknowledge financial support from the US National Science Foundation, US Department of Energy, and the Center for Advanced Process Decision-making at Carnegie Mellon for the research discussed in this paper.

References

1. U. M. Ascher, L. R. Petzold: *Computer Methods for Ordinary Differential Equations and Differential-Algebraic Equations*: SIAM, Philadelphia, PA (1998)
2. R. D. Bartusiak: NLMPC: A platform for optimal control of feed- or product-flexible manufacturing. In: *Assessment and Future Directions of NMPC*, p. 338. Springer, Berlin (2007)
3. J. Betts: *Practical Methods for Optimal Control Using Nonlinear Programming*. SIAM Series on Advances in Design and Control, Philadelphia, PA (2001)
4. H. G. Bock: Numerical treatment of inverse problem in differential and integral equations. In: *Recent Advances in Parameter Identification Techniques for ODE*, Heidelberg, Federal Republic of Germany (1983)
5. J. Busch, M. Oldenburg, J. Santos, A. Cruse, W. Marquardt: Dynamic predictive scheduling of operational strategies for continuous processes using mixed-logic dynamic optimization. *Comput. Chem. Eng.* **31** (2007) pp. 574–587
6. A. M. Cervantes, S. Tonelli, A. Brandolin, J. A. Bandoni, L. T. Biegler: Large-scale dynamic optimization for grade transitions in a low density polyethylene plant. *Computers & Chemical Engineering* **26** (2002) pp. 227–237
7. A. M. Cervantes, A. Wächter, R. Tutuncu, L. T. Biegler: A reduced space interior point strategy for optimization of differential algebraic systems. *Comp. Chem. Engr.* **24** (2000) pp. 39–51
8. M. Diehl, H. G. Bock, J. P. Schlöder, F. Allgöwer, R. Findeisen, Z. Nagy: Real-time optimization and nonlinear MPC of processes governed by differential-algebraic equations. *Journal of Process Control* **12**(4) (2002) pp. 577–585

9. A. V. Fiacco: *Introduction to Sensitivity and Stability Analysis in Nonlinear Programming*: Academic Press, New York (1983)
10. R. Franke, J. Doppelhamer: Integration of advanced model based control with industrial it. In: *Assessment and Future Directions of NMPC*, p. 368. Springer, Berlin (2007)
11. I. E. Grossmann: Enterprise-wide optimization: A new frontier in process systems engineering. *AIChE J.* **51**(7) (2005) pp. 1846–1857
12. M. Grötschel, S. Krumke, J. Rambau (eds.): *Online Optimization of Large Systems*. Springer, Berlin (2001)
13. W. W. Hager: Runge-Kutta methods in optimal control and the transformed adjoint system. *Numer. Math.* **87** (2000) pp. 247–282
14. L. Jiang, L. T. Biegler, V. G. Fox: Simulation and optimization of pressure swing adsorption systems for air separation. *AIChE J.* **49**(5) (2003) pp. 1140–1157
15. J. Kadam, W. Marquardt: Integration of economical optimization and control for intentionally transient process operation In: *Assessment and Future Directions of Nonlinear Model Predictive Control*, R. Findeisen, F. Allgoewer, L. Biegler (eds.) Lecture Notes in Control and Information Sciences **358**, Springer, Berlin/Heidelberg (2007) pp. 419–434
Sensitivity-based solution updates in closed-loop dynamic optimization. In: *Proceedings of the DYCOPS 7 Conference*, Elsevier (2004)
16. 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) pp. 81–126
17. Y. Kawajiri, L. T. Biegler: Optimization strategies for simulated moving bed and powerfeed processes. *AIChE Journal* **52**(4) (2006) pp. 1343–1350
18. C. D. Laird, L. T. Biegler: Large-scale nonlinear programming for multi-scenario optimization. *Modeling, Simulation and Optimization of Complex Processes*, pp. 323–336. Springer-Verlag (2008)
19. L. Magni, R. Scattolini: Robustness and robust design of MPC for nonlinear discrete-time systems. In: *Assessment and Future Directions of NMPC*, vol. 358 of *LNCIS*, pp. 239–254. Springer-Verlag (2007)
20. D. Q. Mayne: Nonlinear model predictive control: challenges and opportunities. In: *Nonlinear Model Predictive Control*, eds. A. Zheng F. Allgöwer, pp. 3–22. Birkhaüser-Verlag, Basel (2000)
21. Z. K. Nagy, R. Franke, B. Mahn, F. Allgöwer: Real-time implementation of nonlinear model predictive control of batch processes in an industrial framework. In: *Assessment and Future Directions of NMPC*, vol. 358 of *LNCIS*, pp. 465–472. Springer-Verlag (2007)
22. M. Hagemann, O. Schenk, A. Wächter: Combinatorial approaches to the solution of saddle-point problems in large-scale parallel interior-point optimization. Tech. Rep. RC 23824, IBM T. J. Watson Research Center, Yorktown, USA (2005)
23. J. Oldenburg, W. Marquardt, Heinz D., D. B. Leineweber: Mixed-logic dynamic optimization applied to batch distillation process design. *AIChE J.* **49**(11) (2003) pp. 2900–2917
24. A. Toumi, M. Diehl, S. Engell, H. G. Bock, J. P. Schlöder: Finite horizon optimizing control of advanced SMB chromatographic processes. In: *16th IFAC World Congress*, Prague, Czech Republic (2005)
25. C. D. Laird, V. M. Zavala, L. T. Biegler: Interior-point decomposition approaches for parallel solution of large-scale nonlinear parameter estimation problems. *Chemical Engineering Science* **63** (2008) pp. 4834–4845
26. V. S. Vassiliadis, R. W. H. Sargent, C. C. Pantelides: Solution of a class of multistage dynamic optimization problems. part i - algorithmic framework. *Ind. Eng. Chem. Res.* **33** (1994) pp. 2115–2123
27. A. Wächter, L.T. Biegler: On the implementation of a primal-dual interior point filter line search algorithm for large-scale nonlinear programming. *Math. Program.* **106**(1) (2006) pp. 25–57
28. V. M. Zavala, L. T. Biegler: The advanced-step NMPC controller: optimality, stability and robustness. *Automatica* **45** (2009) pp. 86–93

29. V. M. Zavala, C. D. Laird, L. T. Biegler: A fast computational framework for large-scale moving horizon estimation. *Journal of Process Control* **18**(9) (2008) pp. 876–884
30. V. M. Zavala L. T. Biegler: Large-scale parameter estimation in low-density polyethylene tubular reactors. *I &EC Research*, **45**(23) (2006) pp. 7867–7881
31. V. M. Zavala, Carl D. Laird, L. T. Biegler: Fast solvers and rigorous models: can both be accommodated in NMPC? *J. Robust and Nonlinear Control* **18**(8) (2008) pp. 800–815