Preprints of the 10th IFAC International Symposium on Dynamics and Control of Process Systems
The International Federation of Automatic Control
December 18-20, 2013. Mumbai, India

# A Survey on Sensitivity-based Nonlinear Model Predictive Control

**Lorenz T. Biegler**

*Chemical Engineering Department, Carnegie Mellon University, Pittsburgh, PA 15213 (e-mail:biegler@cmu.edu).*

Abstract: This survey explores recent results in the development of optimization algorithms and formulations for moving horizon estimation (MHE), nonlinear model predictive control (NMPC) and dynamic real-time optimization (D-RTO), with extrapolative nonlinear dynamic (e.g., first principle) models. We consider Nonlinear Programming (NLP) and NLP sensitivity as natural tools for formulation and efficient solution of optimization problems for these three tasks. For MHE, we develop a maximum likelihood formulation that directly incorporates nonlinear models, and is seamlessly adapted to updating the arrival costs. This approach is also extended easily to M-estimators, which essentially eliminate biased estimates due to gross errors. For NMPC, we develop formulations that incorporate well-understood stability and robustness properties, even for large, complex plant models. Finally, recent work has extended these capabilities from setpoint tracking objectives to more general stage costs that are economically based, thus leading to a robustly stable strategy for D-RTO. In concert with these problem formulations, the realization of MHE, NMPC and D-RTO requires the application of a fast NLP solver for time-critical, on-line optimization, as well as efficient NLP sensitivity tools that eliminate computational delay, and guarantee stability and robustness. Algorithms that meet these demands are explored and outlined for these tasks. Finally, a number of challenging distillation case studies are presented that demonstrate the effectiveness of these optimization-based strategies.

*Keywords:* Nonlinear Programming, Nonlinear Model Predictive Control, Moving Horizon Estimation, Dynamic Real-time Optimization

## 1. INTRODUCTION

Model-based control and optimization frameworks represent an attractive alternative for the operation of complex processes. These frameworks allow the incorporation of accurate and complex dynamic process models and the direct handling of multivariable interactions and operational constraints. In addition, the ability to incorporate nonlinear extrapolative models, including detailed first-principles models, allows tighter integration of the controller to higher level economic optimization layers such as real-time optimization (RTO), planning and scheduling. Important enabling developments for this include advanced strategies for DAE-constrained optimization and NLP algorithms, as well as enhanced formulations with stability and robustness guarantees.

Nonlinear model predictive control for tracking and so-called "economic" stage costs, as well as associated state estimation tasks, are reviewed, formulated and analyzed in considerable detail by Rawlings and Mayne (2009) and Mayne et al. (2000). Due to advances described in Chen and Allgöwer (1998); Magni and Scattolini (2007); Diehl et al. (2005b) and Findeisen et al. (2003), fundamental stability and robustness properties of NMPC are well-known, and many of the key issues related to the applicability and relevance of NMPC are understood. Moreover, the availability of detailed dynamic process models for off-line process analysis and optimization allows NMPC to be realized on challenging process applications. Nevertheless, an important hurdle is the cost and reliability of on-line computation; lengthy and unreliable optimization calculations lead to unsuccessful controller performance.

Recent developments for NMPC address the important problem of computational delay. Li and Biegler (1989) proposed the Newton-type strategy for constrained nonlinear processes, where the nonlinear model is linearized around a nominal trajectory, and a QP is solved at every sampling time. A real-time iteration NMPC was proposed by Diehl et al. (2005a) where only one Newton or QP iteration of the NLP is executed on-line at every sampling time, instead of solving the NLP completely. More generally, NMPC strategies have been developed that separate optimization problem into an off-line NLP based on predicted states, and fast on-line calculation for the actual state. A neighboring extremal update (NEU) approach was proposed in Würth et al. (2009), where an optimal control problem is solved over a long time horizon. Then, during each sampling time a fast update, determined by a QP, is performed for the control variable. Feasibility and optimality criteria are then checked to estimate the performance of each update. This was recently extended in Wolf et al. (2011) with additional QP steps to refine the solution on line to improve controller performance. Also, nonlinear model-predictive control is adjusted online in Alamir (2000) based on the expected computation time and the number of iterations of the optimization problem. Findeisen and Allgöwer (2004) extended the NMPC formulation to account for computational delay. In addition, a number of fast NMPC strategies have been developed including Diehl et al. (2005a) and Ohtsuka (2004).

Knowledge of the plant state is essential for realization of NMPC. In practice state information can only be inferred through a set of noisy measurements, in combination with the dynamic process model. For linear systems this is done via Kalman Filters (KF). Based on linearizations of the nonlinear plant model, Extended Kalman Filters (EKF) are typically applied (Jazwinski (2007); Bryson and Ho (1975)). However, EKF may have poor performance for highly nonlinear systems (Daum (2005); Prakash et al. (2010)), thus spawning related estimation methods that include the Unscented Kalman Filter (Julier et al. (2000)), the Ensemble Kalman Filter (Evensen (1994)), and the Particle Filter (Arulampalam et al. (2002)). On the other hand, none of these methods is able to deal with bounds on the states, and this may lead to increased estimation error or divergence of the estimator (Haseltine and Rawlings (2005)). Remedies to handle constrained nonlinear state estimation include Nonlinear Recursive Dynamic Data Reconciliation (Vachhani et al. (2004)), Unscented Recursive Nonlinear Dynamic Data Reconciliation (Vachhani et al. (2004, 2006)), the Constrained Ensemble Kalman Filter (Prakash et al. (2010)), and the Constrained Particle Filter (Prakash et al. (2008)). In contrast to the above estimators, the state estimation problem can be formulated directly as a nonlinear programming (NLP) problem. Here we consider Moving Horizon Estimation (MHE) (Muske and Rawlings (1993); Michalska and Mayne (1995); Robertson et al. (1996)), which uses a batch of past measurements to find the optimal state estimates with an objective function based on maximum likelihood concepts. MHE has very desirable asymptotic stability properties (Rao et al. (2003)) with bounds on plant states handled directly by the NLP solver. Efficient algorithms for MHE are presented in Ohtsuka and Fujii (1996), Tenny and Rawlings (2002), Zavala et al. (2008), Kuehl et al. (2011) and Abrol and Edgar (2011), which also address computational delay.

Finally, the ability to perform nonlinear state estimation and model-based control naturally extends to dynamic real-time optimization (D-RTO). Current practice in process applications decomposes economic optimization into two layers. First, real-time optimization (RTO), optimizes an economic objective with steady state models, leading to a setpoint handled by the lower-level control layer. The advanced control layer (using, e.g., NMPC) then tracks the setpoint to achieve a new steady state. However, this two-layer structured method assumes that model disturbances and transients are neglected in the RTO layer (Engell (2007); Adetola and Guay (2010)). Moreover, model inconsistency between layers and unresolved transient behavior may lead to unreachable setpoints (Rawlings et al. (2008)). Finally, since the control layer has no information on dynamic economic performance, it may generate trajectories that simply track suboptimal setpoints to steady state (Rawlings and Amrit (2009)).

As a result, it is often desirable to apply economically-oriented NMPC that directly optimizes the plant's economic performance subject to dynamic constraints. Recent studies on dynamic real-time optimization (D-RTO) have reported significant performance improvements with economically-oriented NMPC formulations (Zavala and Biegler (2009b); Rawlings and Amrit (2009); Engell (2007); Aske et al. (2008); Amrit et al. (2013)). In addition, stability theory supporting economically-oriented NMPC requires development beyond the mature results for setpoint tracking based on a discrete Lyapunov analysis. This problem formulation and stability analysis must be modified to ensure a stable and robust D-RTO implementation, especially if steady state operation is expected.

This study addresses recent results for NMPC, MHE and D-RTO that are based on *advanced step* concepts that particularly focus on efficient NLP algorithms for background solutions along with on-line updates based on NLP sensitivity. The next section summarizes both the MHE and NMPC problem formulations used in this study. Section 3 then presents an optimization framework based on interior-point NLP solvers and sensitivity concepts. Section 4 presents properties and formulations for advanced step MHE (asMHE) and NMPC (asNMPC) strategies. Section 5 discusses recent advances to asMHE for the fast calculation of the arrival cost and the incorporation of detection and elimination of measurements with gross errors. Section 6 describes a multi-step extension of asNMPC that allows very large process models to be solved in background over multiple time steps. Section 7 then provides recent updates for economic NMPC properties and demonstrates their impact with a large-scale D-RTO study. All of these advances are demonstrated with large-scale distillation case studies with nonlinear first principle models. Finally, section 8 summarizes the paper along with directions for future work.

## 2. NLP STRATEGIES FOR MHE AND NMPC

We begin with the following discrete-time nonlinear dynamic model of the plant:

$$x_{k+1} = f(x_k, u_k) + \xi_k, \qquad y_k = h(x_k) + \zeta_k \tag{1}$$

where $x_k \in \Re^{n_x}$ is the *true* plant state at sampling time $t_k$ and $u_k \in \Re^{n_u}$ is the control input. The nonlinear dynamic model $f(\cdot,\cdot) : \Re^{n_x+n_u} \to \Re^{n_x}$ is the nominal model. The observed output $y_k \in \Re^{n_y}$ with $n_y \leq n_x$ is related to the state-space $x_k$ through the nonlinear mapping $h(\cdot) : \Re^{n_x} \to \Re^{n_y}$. The true plant deviates from the nominal prediction due to the process disturbance $\xi_k \in \Re^{n_x}$ and measurement noise $\zeta_k \in \Re^{n_y}$ which we assume are Gaussian errors with $\xi_k \sim \mathcal{N}(0, Q_k)$ and $\zeta_k \sim \mathcal{N}(0, R_k)$.

Using this information, we would like to compute an estimate $\tilde{x}_k$ of the current state $x_k$ that can be used for our model-based controller. In order to do this, we first solve the following MHE problem:

$$\min_{z_{k-N},\cdots,z_k} \; \Phi(z_{k-N}) + \frac{1}{2}\Big(\sum_{l=k-N}^{k} \zeta_l^T R_l^{-1} \zeta_l + \sum_{l=k-N}^{k-1} \xi_l^T Q_l^{-1} \xi_l\Big) \tag{2a}$$

$$\text{s.t. } z_{l+1} = f(z_l, u_l) + \xi_l, \; l = k-N, \ldots, k \tag{2b}$$

$$y_l = g(z_l) + \zeta_l \tag{2c}$$

$$z_l \in \mathbb{X} \tag{2d}$$

where $\{z^*_{k-N|k}, \ldots, z^*_{k|k}\}$ refers to the optimal state estimates at each time step $k$ and $\Phi(z_{k-N}) = ||z_{k-N} - \bar{x}_{k-N}||^2_{\hat{\Pi}^{-1}_{k-N|k-1}}$ is the arrival cost that represents all the information before $k-N$, not included in the horizon. Here the prior estimate of the past plant state is denoted as $\bar{x}_{k-N}$ with its associated covariance $\hat{\Pi}^{-1}_{k-N|k-1}$.

From the solution of (2) we obtain the optimal estimate $\tilde{x}_k = z^*_{k|k}$ and define the NMPC problem,

$$\min_{v_l} \quad \Psi(z_N) + \sum_{l=0}^{N-1} \psi(z_l, v_l) \tag{3a}$$

$$\text{s.t. } z_{l+1} = f(z_l, v_l) \quad l = 0, \dots N-1 \tag{3b}$$

$$z_0 = \tilde{x}_k \tag{3c}$$

$$z_l \in \mathbb{X}, v_l \in \mathbb{U}, z_N \in \mathbb{X}_f \tag{3d}$$

where we assume that the states and controls are restricted to the domains $\mathbb{X}$ and $\mathbb{U}$, respectively. The stage cost is given by $\psi(\cdot,\cdot) : \Re^{n_x+n_u} \rightarrow \Re$, while the terminal cost is denoted by $\Psi(\cdot) : \Re^{n_x+n_u} \rightarrow \Re$. For tracking problems, we can assume that the states and control variables can be defined with respect to setpoint and reference values, and that when $\zeta = \xi = 0$ the dynamic model (1) has the property: $f(0,0) = 0, g(0) = 0$.

After solution of (3) the control action is extracted from the optimal trajectory $\{z_0^* ... z_N^* \, v_0^*, ..., v_{N-1}^*\}$ as $u_k = v_0^*$ At the next time, the plant evolves as,

$$x_{k+1} = f(x_k, u_k)) + \xi_k, \; y_{k+1} = g(x_{k+1}) + \zeta_{k+1} \tag{4}$$

and we shift the measurement sequence one step forward to solve the new MHE problem (2) and obtain $\tilde{x}_{k+1}$. Then we set $k = k+1$ and use the new state estimate $\tilde{x}_k$ to solve the next NMPC problem (3).

We refer to the above strategies as *ideal MHE* and *ideal NMPC*, where the on-line calculation time is neglected. Ideal NMPC has well-known stability properties (see Magni and Scattolini (2007); Zavala and Biegler (2000)). with the following assumptions.

*Definition 1.* A continuous function $\alpha(\cdot) : \Re \rightarrow \Re$ is a $\mathscr{K}$ function if $\alpha(0) = 0, \alpha(s) > 0, \forall s > 0$ and it is strictly increasing.

*Assumption 1.* (Nominal Stability Assumptions of Ideal NMPC)

(1) The terminal cost $\Psi(\cdot)$ satisfies $\Psi(x) > 0, \forall x \in \mathbb{X}_f \setminus \{0\}$.
(2) There exits a local control law $u = h_f(x)$ such that $f(x, h_f(x)) \in \mathbb{X}_f, \forall x \in \mathbb{X}_f$.
(3) $\Psi(f(x, h_f(x))) - \Psi(x) \leq -\psi(x, h_f(x)), \forall x \in \mathbb{X}_f$.
(4) $\psi(x,u)$ satisfies $\alpha_p(|x|) \leq \psi(x,u) \leq \alpha_q(|x|)$ where $\alpha_p(\cdot)$ and $\alpha_q(\cdot)$ are $\mathscr{K}$ functions.

where $\mathbb{X}_f$ is the terminal region.

Under these assumptions we can state the following theorem.

*Theorem 1.* (Nominal Stability of Ideal NMPC) Consider the moving horizon problem (3) and associated control law $u = h_f(x)$ that satisfies Assumption 1. Then, the objective function of (3) is a Lyapunov function and the closed-loop system is asymptotically stable.

Robust stability properties for NMPC are developed in Rawlings and Mayne (2009). In particular, Ideal NMPC approach satisfies the *Input to State Stability (ISS)* property for plants described by (1) that have bounded uncertainties.

On the other hand, the computation time to solve (2) and (3) does lead to computational feedback delay, which can impact plant performance and even destabilize the process. To reduce the on-line computational time we consider two crucial elements, a fast NLP algorithm for background solution of the NLPs (2) and (3) and a fast on-line correction step based on NLP sensitivity. In the next section, we develop both tools in the context of primal-dual interior-point solvers.

## 3. INTERIOR-POINT NLP SOLVERS

Both NLP problems (2) and (3) can be represented as:

$$min_{\mathbf{x}} \, F(\mathbf{x}, \eta), \; s.t. \; c(\mathbf{x}, \eta) = 0, \; x \geq 0 \tag{5}$$

where $\mathbf{x} \in \Re^n$ is the variable vector containing the states, controls and outputs, and $\eta$ is a *fixed* data vector used for sensitivity purposes. The equality constraints are $c(\mathbf{x}, \eta) : \Re^n \rightarrow \Re^m$. In interior-point solvers, the inequality constraints of problem (5) are handled *implicitly* by adding barrier terms to the objective function,

$$min_{\mathbf{x}} \, F(\mathbf{x}, \eta) - \mu \sum_{j=1}^{n_{\mathbf{x}}} \ln(\mathbf{x}^{(j)}), \tag{6a}$$

$$s.t. \qquad \mathbf{c}(\mathbf{x}, \eta) = 0, \tag{6b}$$

where $\mathbf{x}^{(j)}$ denotes the $j^{th}$ component of vector $\mathbf{x}$. Solving (6) for the sequence of $\mu^l \rightarrow 0$, with $l = 0, 1, 2, \dots, \infty$ leads to solution of the original NLP (5). As shown in Forsgren et al. (2002), convergence of solutions of (6) to (5) have been proved under mild conditions.

For a given barrier parameter value $\mu$, IPOPT solves the primal-dual optimality conditions of barrier problems (6) directly,

$$\nabla_{\mathbf{x}} F(\mathbf{x}, \eta) + \nabla_{\mathbf{x}} \mathbf{c}(\mathbf{x}, \eta) \lambda - \nu = 0, \tag{7a}$$

$$\mathbf{c}(\mathbf{x}, \eta) = 0, \tag{7b}$$

$$\mathbf{X} \cdot \mathbf{V} e = \mu e, \tag{7c}$$

where $\mathbf{X} = \text{diag}(\mathbf{x}), \mathbf{V} = \text{diag}(\nu)$, $e \in \Re^{n_{\mathbf{x}}}$ is a vector of ones, and $\lambda \in \Re^{n_\lambda}$ and $\nu \in \Re^{n_{\mathbf{x}}}$ are Lagrange multipliers for the equality constraints and bounds, respectively. The gradient of the objective function is $\nabla_{\mathbf{x}} F(\mathbf{x}, \eta) \in \Re^{n_{\mathbf{x}}}$ while $\nabla_{\mathbf{x}} \mathbf{c}(\mathbf{x}, \eta) \in \Re^{n_{\mathbf{x}} \times n_\lambda}$ is the constraint Jacobian. To solve this system of nonlinear equations IPOPT uses an exact Newton method, starting the iteration sequence at point $s_o^T := [\mathbf{x}_o^T \, \lambda_o^T \, \nu_o^T]$. At the $i$th Newton iteration, the search direction $\Delta s_i = s_{i+1} - s_i$ is computed by linearization of the KKT conditions (7),

$$\begin{bmatrix} \mathbf{H}_i & \mathbf{A}_i & -I \\ \mathbf{A}_i{}^T & 0 & 0 \\ \mathbf{V}_i & 0 & \mathbf{X}_i \end{bmatrix} \begin{bmatrix} \Delta \mathbf{x}_i \\ \Delta \lambda_i \\ \Delta \nu_i \end{bmatrix} = - \begin{bmatrix} \nabla_{\mathbf{x}} f(\mathbf{x}_i, \eta) + \mathbf{A}_i \lambda_i - \nu_i \\ \mathbf{c}(\mathbf{x}_i, \eta) \\ \mathbf{X}_i \mathbf{V}_i e - \mu e \end{bmatrix} \tag{8}$$

where $\mathbf{A}_i := \nabla_{\mathbf{x}} \mathbf{c}(\mathbf{x}_i, \eta)$. Matrix $\mathbf{H}_i \in \Re^{n_{\mathbf{x}} \times n_{\mathbf{x}}}$ is the Hessian of the Lagrange function $\mathscr{L} = F(\mathbf{x}_i, \eta) + \mathbf{c}(\mathbf{x}_i, \eta) \lambda_i - \nu_i{}^T \mathbf{x}_i$. After solving a sequence of barrier problems for $\mu \rightarrow 0$, the solver returns the solution triple $s_*^T(\eta) = [\mathbf{x}_*^T \, \lambda_*^T \, \nu_*^T]$ for problem (5).

Solving the KKT system (8) is the most computationally demanding step in the solution of the NLP. However, the KKT matrix arising from discretized dynamic optimization problems is often very sparse and structured. In IPOPT, after eliminating the bound multipliers from the KKT system (8), we apply a symmetric indefinite factorization of the resulting KKT matrix. The computational complexity of this strategy is generally favorable (often scaling nearly linearly with problem size). However, significant fill-in and computer memory bottlenecks might arise during the factorization step if the sparsity pattern is not properly exploited. In order to factorize the KKT matrix, efficient sparse linear solvers should be applied, e.g. HSL library with the METIS option (Duff (2004)). Moreover, since the structure of the KKT matrix does not change between iterations, the linear solver needs to analyze the sparsity pattern *only once*.

For sensitivity of the NLP, we note that problem (5) is parametric in the data $\eta$ and the optimal primal and dual variables can be treated as implicit functions of $\eta$. For a *sufficiently small* $\mu_\ell$, the KKT conditions (7) of the barrier problem (6) can be expressed as the equations $\varphi(s(\eta),\eta) = 0$ and we denote $\mathbf{K}_*(\eta_0)$ as the KKT matrix in (8).

To compute approximate neighboring solutions around an already available nominal solution $s_*(\eta_0)$, we invoke the following classical results,

*Theorem 2.* (NLP Sensitivity) (Fiacco (1983)) Assume that $F(\cdot)$ and $c(\cdot)$ of the parametric problem (5) are twice continuously differentiable in a neighborhood of the nominal solution $s_*(\eta_0)$ and this solution satisfies LICQ, strict complementarity and SSOC, then $s_*(\eta_0)$ is an isolated local minimizer of (5) and the associated Lagrange multipliers are unique. Moreover, for $\eta$ in a neighborhood of $\eta_0$ there exists a unique, continuous and differentiable vector function $s_*(\eta)$ which is a local minimizer satisfying SSOC, strict complementarity and LICQ for (5). Finally, there exists a positive Lipschitz constant $L$ such that $\|s_*(\eta) - s_*(\eta_0)\| \le L\|\eta - \eta_0\|$ along with a positive Lipschitz constant $L_F$ such that the optimal values $F(\eta)$ and $F(\eta_0)$ satisfy $\|F(\eta) - F(\eta_0)\| \le L_F\|\eta - \eta_0\|$.

Under these results, a step $\Delta s(\eta)$ computed from,

$$\begin{aligned}\mathbf{K}_*(\eta_0)\Delta s(\eta) &= -\left(\varphi(s_*(\eta_0),\eta) - \varphi(s_*(\eta_0),\eta_0)\right) \\ &= -\varphi(s_*(\eta_0),\eta). \end{aligned} \tag{9}$$

with $\Delta s(\eta) = \tilde{s}(\eta) - s_*(\eta_0)$, is a Newton step taken from $s_*(\eta_0)$ towards the solution of a neighboring problem. Consequently, $\tilde{s}(\eta)$ satisfies,

$$\|\tilde{s}(\eta) - s_*(\eta)\| \le L_s\|\eta - \eta_0\|^2 \tag{10}$$

with $L_s > 0$. Furthermore, since the KKT matrix $\mathbf{K}_*(\eta_0)$ is already available from the solution of the nominal problem, computing this step requires only a *single backsolve*, which can be performed orders of magnitude faster than the factorization of the KKT matrix.

When the perturbation $\eta - \eta_0$ induces an active-set change, the linearization of the complementarity relaxation (7c) contained in the nominal KKT matrix $\mathbf{K}_*(\eta_0)$ may drive the first Newton iteration *outside* of the feasible region and the sensitivity approximation is inconsistent. A number of strategies have been developed to accommodate active set changes. These include the extension of (9) to quadratic programming problems as well as simple clipping strategies, which choose a stepsize along $\Delta s$ to ensure that $u_k$ remains within bounds. Detailed discussion and evaluation of these strategies can be found in Yang and Biegler (2013).

## 4. ADVANCED-STEP MHE AND NMPC STRATEGIES

Treatment of problems (2) and (3) with the above NLP and sensitivity tools corresponds to the off-line and on-line components, respectively, of advanced step strategies. At time $t_k$ we use the current estimate $\tilde{x}_k$ and control $u_k$ to predict the future state and associated measurement,

$$\bar{x}_{k+1} = f(\tilde{x}_k, u_k), \qquad \bar{y}_{k+1} = g(\bar{x}_{k+1}) \tag{11}$$

With the prediction $\bar{y}_{k+1}$ we begin the execution of the MHE problem (2). Simultaneously, we can use the predicted state $\bar{x}_{k+1}$ to start the execution of the NMPC problem (3). Since both problems are decoupled their executions can be done in parallel, thus reducing solution time. At the solution of these problems, we hold the corresponding KKT matrices $\mathbf{K}_*^{mhe}$ and $\mathbf{K}_*^{mpc}$.

Once the true measurement $y_{k+1}$ becomes available, we compute a fast backsolve with $\mathbf{K}_*^{mhe}$ to obtain an *approximate* state estimate $\tilde{x}_{k+1}^{as}$ (different from the optimal state estimate $\tilde{x}_{k+1}$). Using the approximate state estimate we perform a fast backsolve with $\mathbf{K}_*^{mpc}$ to obtain the approximate control action $u_{k+1}$. Consequently, the proposed framework for the advanced-step MHE and NMPC strategies, *asMHE* and *asNMPC*, respectively, can be summarized as follows:

In background, between $t_k$ and $t_{k+1}$:

(1) Use *current* estimate $\tilde{x}_k^{as}$ and control $u_k$ to predict the future state $\bar{x}_{k+1} = f(\tilde{x}_k^{as}, u_k)$ and corresponding output measurement $\bar{y}_{k+1} = g(\bar{x}_{k+1})$.
(2) Using the predicted $\bar{y}_{k+1}$ and $\bar{x}_{k+1}$ solve problems (2) and (3) between $t_k$ and $t_{k+1}$.
(3) Hold the KKT matrices $\mathbf{K}_*^{mhe}$ and $\mathbf{K}_*^{mpc}$.

On-line, at $t_{k+1}$:

(1) Obtain the true measurement $y_{k+1}$ and reuse factorization of $\mathbf{K}_*^{mhe}$ to quickly compute $\tilde{s}_{MHE}$ from (9) and extract $\tilde{x}_{k+1}^{as}$.
(2) Use $\tilde{x}_{k+1}^{as}$ and reuse factorization of $\mathbf{K}_*^{mpc}$ to quickly compute $\tilde{s}_{MPC}$ from (9) and extract $u_{k+1}$.
(3) Set $k := k+1$, and return to background.

Stability and robustness properties of the advanced step strategy have been analyzed in Zavala and Biegler (2009a). Since explicit bounds on the estimator error $\|x_k - \tilde{x}_k\|$ can be established for the MHE formulation (2) (Alessandri et al. (2008)), this error can be treated as a disturbance $\xi_k$ with $\tilde{x}_k := x_k + \xi_k$. The following robustness result applies to the combined *asMHE* and *asNMPC* strategies.

*Theorem 3.* Assume that the NLPs for (2) and (3) can be solved within one sampling time. Assume also that nominal and robust stability assumptions for ideal NMPC hold (see Zavala and Biegler (2009a)), then there exist bounds on the noise $\xi$ and $\zeta$ for which the objective function, obtained from the combined asMHE-asNMPC strategy, is an input to state stable (ISS) Lyapunov function, and the resulting closed-loop system is ISS stable.

### 4.1 Distillation Case Study

To demonstrate the performance of the asNMPC controller, we consider a distillation example that separates methanol and n-propanol. The model originates from Diehl et al. (2002) and consists of differential-algebraic equations (DAEs) based on dynamic MESH (mass, equilibrium, summation and heat) equations for each tray as well as condenser (i=0) and reboiler (i=NT+1). Here we consider a column with $N_T = 40$ trays along with a total condenser and a reboiler. The feed stream enters the column at tray $i = 21$. Detailed presentation of the DAE model, along with physical property equations (enthalpy, Antoine equation, etc.) can be found in Diehl et al. (2002) and López-Negrete et al. (2013).

For this example we considered 60 second sampling times, and 10 sampling times in the predictive horizon. The continuous time DAE model is transformed into the discrete time model (1) using orthogonal collocation on finite elements. Using 3

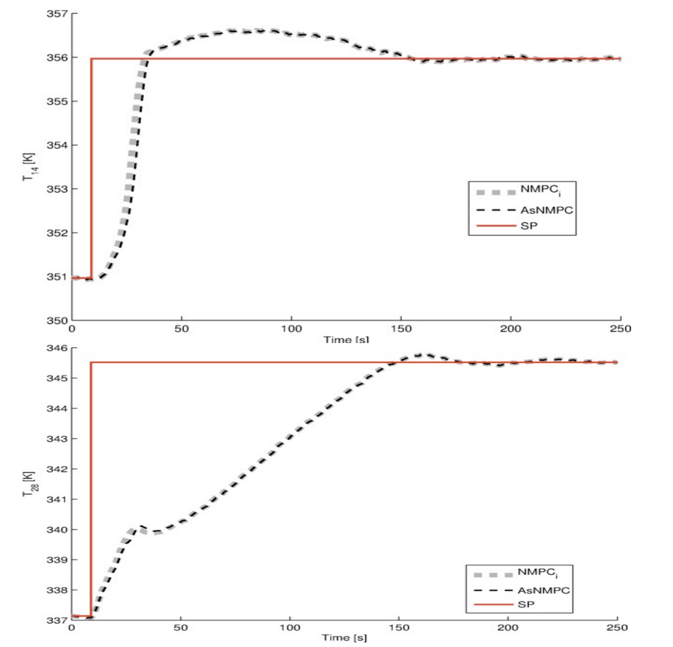


Figure 1. asNMPC control of a binary distillation column showing Temperatures on trays 14 and 28

point Radau collocation, the NLP consists of 19814 variables and 19794 equality constraints. To account for plant model mismatch we add zero mean, Gaussian noise to the differential variables (total molar holdup and liquid compositions at each tray) with variance $10^{-4}$ for the holdups and $10^{-6}$ for the compositions. The control variables of this problem are the reboiler heat duty and the reflux ratio. The objective function takes the form in (3) with two setpoints on temperatures from trays 14 and 28. These two measurements are sensitive to changes in overhead and bottom product compositions.

Figure 1 shows the simulation results for the change from setpoint 1 to setpoint 2, and compares the performance of advanced step NMPC with the ideal case. The average solution time of the NLP (3) was 9.4 CPU seconds, while the sensitivity step (9) takes an average of 0.063 CPU seconds. Thus, computational delay has been reduced by over two orders of magnitude. Both ideal and advanced step NMPC strategies effect the setpoint change and show virtually identical performance.

## 5. ADVANCES IN MOVING HORIZON ESTIMATION

NMPC formulations are readily adapted to the integration of state estimation components. In previous work, we analyzed the application of fixed gain and Extended Luenberger observers, along with EKF. Incorporating these state estimators within NMPC leads to input to output practical stability (ISpS) strategies (Huang et al. (2010b, 2012b)). Moreover, as demonstrated in Huang et al. (2010a), asNMPC combined with these observers is straightforward to implement and leads to efficient, offset-free performance even on challenging nonlinear process systems, including air separation units with over 1500 DAEs.

These formulations are very successful for combined state estimation and nonlinear control. On the other hand, accurate distributional information on these estimates is often lacking. As an alternative, the more complex application of MHE requires solution of an additional NLP (2), but (with Gaussian $\xi_l, v_l$) also provides a firm statistical basis for the estimates.

Moreover, the advanced step MHE formulation allows for the ready integration of a consistent state estimation strategy with asNMPC, with negligible additional on-line cost; an approximate state estimate $\tilde{x}(k)$ can be obtained with a fast on-line computation requiring a single backsolve of (9). In addition, after new measurements are obtained at $t_k$ the arrival cost $\Phi(\tilde{x}_{k-N})$ in (2) requires an update of the covariance $\hat{\Pi}^{-1}_{k-N|k-1}$. Detailed derivation of this covariance update is given in López-Negrete and Biegler (2012). A prior covariance estimate is generated by propagating the EKF equations forward, while the posterior covariance is derived from an Extended Kalman Smoother (EKS). Both updates arise from discrete Riccati equations that grow cubically with problem size; this covariance update becomes expensive for large systems.

Instead, the posterior update can be computed very efficiently from the sensitivity of the KKT system of (2). In López-Negrete and Biegler (2012) we prove the following result.

*Theorem 4.* (Covariance of state estimates). Assume that the solution of (2) has no active bound constraints, and the linear independence constraint qualification (LICQ) and sufficient second order conditions hold. Then the inverse of the reduced Hessian (2), with the choice of $z_{k-N}$ as the independent variables, is an approximation of the smoothed covariance for the arrival cost.

Moreover, for nonlinear dynamic processes, the reduced Hessian approximation of the covariance matrices incorporates additional second order information, which is not included in typical EKS (or EKF) formulations. Otherwise, the following result from López-Negrete and Biegler (2012) shows where the reduced Hessian covariance is equivalent to the EKS formulation.

*Theorem 5.* (Covariance of linear Gaussian systems). For a linear unconstrained Gaussian system the approximation given in Theorem 4 is exact and the inverse of the reduced Hessian is the Kalman Smoothing covariance.

Extracting reduced Hessian information from (2) can be done very efficiently from the KKT conditions of the NLP (Pirnay et al. (2012)). To see this, we modify (5) as:

$$\min F(\mathbf{x}), \text{ s.t. } c(\mathbf{x}) = 0 \tag{12}$$

and assume that no bounds on $\mathbf{x}$ are active at the solution. Furthermore, we partition the variable vector into dependent ($\mathbf{x}_D \in \mathbb{R}^m$) and independent ($\mathbf{x}_I \in \mathbb{R}^{n-m}$) variables, $\mathbf{x}^T = [\mathbf{x}_D^T, \mathbf{x}_I^T]$. We then modify the KKT system of (12) as follows:

$$\begin{bmatrix} \mathbf{H}^* & \mathbf{A}^* \\ \mathbf{A}^{*,T} & 0 \end{bmatrix} \begin{bmatrix} S_x \\ S_\lambda \end{bmatrix} = -\begin{bmatrix} T \\ 0 \end{bmatrix} \tag{13}$$

where the matrix $T \in \mathbb{R}^{(n-m)\times n}$ is given by $T^T = [0 \mid I_{n-m}]$. We define $S_x = ZS_Z + YS_Y$, where $Z \in \mathbb{R}^{(n-m)\times n}$, $Y \in \mathbb{R}^{m\times n}$, $S_Z \in \mathbb{R}^{(n-m)\times(n-m)}$, $Y \in \mathbb{R}^{m\times(n-m)}$ and choose $A^{*,T}Z = 0$, with $A^{*,T}Y$ and $R = [Y \mid Z]$ nonsingular. Also, we determine $Z$ and $Y$ by using $A^{*,T} = [A_D^T \mid A_I^T]$, with $A_D$ square and nonsingular. This leads to $Y^T = [I_m \mid 0]$, $Z^T = [-A_I(A_D)^{-1} \mid I_{n-m}]$.

Multiplying the first row of (13) by $R^T$ and substituting $S_x = ZS_Z + YS_Y$ leads to the equivalent linear system

$$\begin{bmatrix} Y^T\mathbf{H}^*Y & Y^T\mathbf{H}^*Z & Y^T\mathbf{A}^* \\ Z^T\mathbf{H}^*Y & Z^T\mathbf{H}^*Z & 0 \\ \mathbf{A}^{*,T} & 0 & 0 \end{bmatrix} \begin{bmatrix} S_Y \\ S_Z \\ S_\lambda \end{bmatrix} = -\begin{bmatrix} Y^T T \\ Z^T T \\ 0 \end{bmatrix} \tag{14}$$

From (14) we have $S_Y = 0$ and $S_Z = (Z^T \mathbf{H}^* Z)^{-1}$, the inverse of the reduced Hessian. Consequently, with the choice of $z_{k-N}$ as the independent variables, this sensitivity-based approach leads to efficient determination of the smoothed covariance for the arrival cost in (2). Moreover, compared to the classical forward/backward (EKF/EKS) evolution of the covariance with the Riccati equations, with cubic complexity, we see that the computational cost increases approximately linearly with our approach. For a system with 250 states, this leads to a *500-fold reduction* in the calculation of the covariance matrix (López-Negrete (2011)).

### 5.1 M-Estimator MHE Formulation for Gross Errors

Using asMHE for state estimation relies on predicted measurements that are close to the actual measurements. However, when measurements are contaminated with gross errors we also need to assess their influence on the accuracy of the state estimates. For this, Nicholson et al. (2013) develop an extended MHE formulation that is robust to gross errors. Detecting such errors and eliminating their bias is a crucial step to obtaining robust state estimates. Robust M-Estimators for gross error detection are considered here because they can easily be implemented within our NLP formulation.

Two robust M-Estimators are considered in Nicholson et al. (2013), the Fair (Huber-type) and Redescending (Hampel-type) estimators. The Fair Function (Huber (1981)) is given by

$$\rho_j^F(\varepsilon_j) = C^2 \left[ \frac{|\varepsilon_j|}{C} - \log\left(1 + \frac{|\varepsilon_j|}{C}\right) \right] \tag{15}$$

where $\rho_j$ is the estimator associated with the $j^{\text{th}}$ measurement, $C$ is a tuning parameter, and $\varepsilon_j$ is the studentized prediction error. The influence function for the Fair Function is proportional to its first derivative:

$$\frac{d\rho_j^F(\varepsilon_j)}{d\varepsilon_j} = \frac{\varepsilon_j}{1 + \frac{|\varepsilon_j|}{C}} \tag{16}$$

For small residuals the Fair Function is a good approximation of the least squares estimator, but as residuals become larger it transitions to a linear function where less weight is put on gross error measurements. Also, the influence function of the Fair Function will approach a constant, confirming that the Fair Function is robust to large measurement errors. Finally, the Fair Function can be adjusted using its tuning parameter $C$ to balance efficiency with robustness. Smaller values of $C$ cause the Fair Function to become less efficient, thus causing smaller residuals to deviate more from a Gaussian distribution.

The Redescending estimator (Hampel (1974)) is given by the piecewise function:

$$\rho_j^R(\varepsilon_j) = \begin{cases} \frac{1}{2}\varepsilon_j^2, & |\varepsilon_j| \in [0,a) \\ a|\varepsilon_j| - \frac{a^2}{2}, & |\varepsilon_j| \in [a,b) \\ \frac{2ab-a^2}{2} + \frac{a(b^2 - 2(b+|\varepsilon_j|)c + |\varepsilon_j|^2)}{2(c-b)}, & |\varepsilon_j| \in [b,c) \\ ab - \frac{a^2}{2} + \frac{a(c-b)}{2}, & |\varepsilon_j| \geq c, \end{cases} \tag{17}$$

where $\varepsilon_j$ is again the studentized prediction error and $a$, $b$, and $c$ are tuning parameters, which define the 4 regions in the estimator. Each region behaves differently depending on the magnitude of the residuals. The first derivative of the Redescending estimator is

$$\frac{d\rho_j^R(\varepsilon_j)}{d\varepsilon_j} = \begin{cases} \varepsilon_j, & |\varepsilon_j| \in [0,a) \\ \pm a, & |\varepsilon_j| \in [a,b) \\ \frac{\pm a(c-|\varepsilon_j|)}{(c-b)}, & |\varepsilon_j| \in [b,c) \\ 0, & |\varepsilon_j| \geq c. \end{cases} \tag{18}$$

Again, for small residuals the Redescending estimator is a good approximation of the least squares estimator. As the residuals increase, they have less influence on $\rho^R$ until eventually the estimator just becomes a constant value. This translates to an influence function that becomes zero after passing the threshold defined by the tuning parameter $c$. In practice, this has the same effect as removing any gross error measurement that exceeds this threshold. The Redescending estimator depends on three tuning parameters. Heuristic tuning methods (Arora and Biegler (2001)) have been developed and good performance can be obtained from either the Fair Function or the Redescending estimator for various tuning parameter values.

M-estimators can be incorporated into our NLP formulation by simply replacing the measurement error terms in the objective function with the $\rho$ function of the desired estimator. Our new NLP problem becomes

$$\min_{z_{k-N},\dots,z_k} \Phi(z_{k-N}) + \frac{1}{2}\left( \sum_{l=k-N}^{k} \rho_l^{ME}(v_l) + \sum_{l=k-N}^{k-1} \xi_l^T Q_l^{-1} \xi_l \right) \tag{19a}$$

$$\text{s.t. } z_{l+1} = f(z_l, u_l) + \xi_l \tag{19b}$$

$$y_l = g(z_l) + v_l \tag{19c}$$

$$z_l \in \mathbb{X} \tag{19d}$$

One danger of using robust M-Estimators is the possibility of type-1 errors, i.e. incorrectly identifying good measurements as gross errors. While there is no way to completely eliminate type 1 errors there is a trade-off that must be considered between the influence put on large residuals and the number of type-1 errors. This trade-off must be taken into account when choosing the tuning parameters for the robust estimators in order to achieve good state estimation. Another concern is nonconvexity of the Redescending estimator. Even with nonlinear plant models, the likelihood of obtaining local solutions (and Type 1 errors) is higher with the Redescending estimator, and careful initialization strategies should be considered for (19).

### 5.2 Distillation Case Study

For our state estimation formulations we use a horizon length of 10 measurements and we discretize the model using orthogonal collocation with 3 collocation points. NLP (19) now has 21,642 variables and 20,718 constraints. The simulations are run for 300 time steps and each sampling time is 60 seconds. Again we assume that there is no plant-model mismatch in order to isolate the effects of the gross errors on each formulation. The covariance values for the measurements on tray temperature and volumetric tray holdup are $diag\{0.0625, 10^{-8}\}$ while covariance for the process noise on mole fraction, tray molar holdup, condenser holdup and reboiler holdup are $diag\{10^{-5}, 1, 10, 5\}$.

In general, we see that the MHE and asMHE estimates are very similar but the average on-line time required for the asMHE

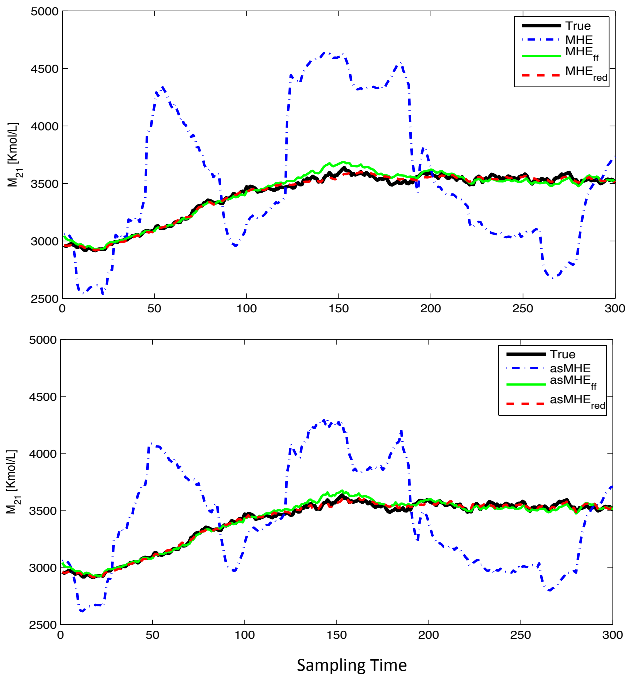


Figure 2. Comparison of the MHE and asMHE formulations with and without M-Estimators for the distillation case, showing $M_{21}$, the molar holdup on tray 21

formulation is only 0.022% of the time required to solve the MHE problem. The MHE formulation (2) took on average 45.48 CPU seconds to solve, while the on-line component of the asMHE formulation requires an average of 0.01 CPU seconds. This leads to a large reduction in the on-line computational expense without sacrificing the performance of the estimator.

We generate gross errors as sudden step changes in the measurement errors in the temperature measurements and volumetric holdups on 20 of the trays. Magnitude and location of the step errors were generated randomly and about 15% of all measurements contain gross errors. The results are shown in Figure 2. We observe that the two M-estimators provide a significant improvement over the state estimates, when compared to MHE with a weighted least squares formulation. Here the normalized sum of squared errors for the state estimates are 864.43 for MHE with least squares and 594.62 for asMHE. Using the Fair function, the state estimate errors reduce to 2.27 for MHE and 1.86 for asMHE, while for the Redescending function, the state estimate errors are 0.962 for MHE and 0.818 for asMHE. Thus, while MHE and asMHE estimators based on (2) are unable to track the true state, the Fair and Redescending modifications, based on (19) are still able to predict accurate state estimates.

## 6. THE amsNMPC FORMULATION

The asNMPC strategy requires that problem (3) be solved within one sampling time. If solution of problem (3) requires $N_{samp} \geq 1$ sampling times, one could clearly update the control variables less frequently, and thus detune the controller, but performance of the controller would certainly suffer. For this case, Yang and Biegler (2013) instead propose *advanced multi-step NMPC* (amsNMPC) methods, which take two forms. The serial approach performs a sensitivity update of the control variable at every sampling time but solves the NLP problem (3) over $N_{samp} \geq 1$ sampling times. On the other hand, the parallel approach is applied over $N_{samp}$ processors and initiates the solution of problem (3) at every sampling time. When the controller receives an updated state estimate, the solution of the most recent NLP solution receives a sensitivity correction (9) to obtain the corresponding control variable, and the free processor is applied to initiate a new NLP problem. The cycle then repeats as NLP solutions and free processors are obtained. The algorithms proceed as follows.

*Parallel Approach*

When $N_{samp}$ sampling times are required to solve the NLP problem, we solve problem (3) $N_{samp}$ sampling times in advance to get the control variable for the current state. Here, $v_{0|k}$ is updated from (9) once the new state estimate $\tilde{x}_k$ is obtained and we define $z_{N_{samp}|k}$ as the $N_{samp}$th predicted state from the NLP solution, given the initial condition $\tilde{x}_k$. The updated $z_{N_{samp}|k}$ is also the prediction that becomes the initial value of the next NLP problem. Once the control variable is updated through (9) at $t_{k+1}$, the next available processor is applied to deal with the next background NLP (3). The parallel approach is implemented as follows:

For $i = 0 : N_{samp} - 1$,

*Online*: at $t_{k+i}$, having $\tilde{x}_{k+i}$, update $v_{0|k+i}$ and $z_{N_{samp}|k+i}$ using $(s^*(p_0) + \Delta s(p))$ from (3). Inject the updated $v_{0|k+i}$ as $u(k+i)$ to the plant.

*Background*: take the updated $z_{N_{samp}|k+i}$ as the initial value and solve the NLP problem (3) using a new processor.

Set $k = k + N_{samp}$ and repeat the cycle.

*Serial Approach*

Again, $N_{samp}$ sampling times are needed to solve the NLP problem and the solution of the last NLP problem is known at $t_k$. Suppose the optimal solution of the last NLP solution is known at $t_k$. Knowing $\tilde{x}_k$ and $u_k$ we predict $z_{N_{samp}|k}$ as an initial value and solve problem (3) between $t_k$ and $t_{k+N_{samp}}$. In the meantime, the current control variables $v_1, v_2, \ldots, v_{N_{samp}-1}$ are updated online using the sensitivity $(s^*(p_0) + \Delta s(p))$, based on solution of the previous NLP problem. However, for these sensitivity-based updates, (9) cannot be applied directly. Instead an augmented sensitivity system is constructed to account for the predicted states and controls between $t_k$ and $t_{k+N_{samp}}$. This augmented KKT sensitivity system is solved via Schur complement decomposition and is described in Yang and Biegler (2013).

The serial amsNMPC strategy is executed as follows:

*Background*: At $t_k$, having $\tilde{x}_k$ and $u_k$, update $v_0$ and $z_{N_{samp}}$ using $\Delta s$ from (9). Solve problem (3) between $t_k$ and $t_{k+N_{samp}}$ with an updated $z_{N_{samp}}$ as the initial value.

*Online*: for $i = 1 : N_{samp} - 1$,

At $t_{k+i}$, having $z(k+i)$, update the augmented sensitivity system and obtain $v_i$ using $\Delta s$. Inject the updated $v_i$ as $u(k+i)$ to the plant.

Set $k = k + N_{samp}$ and repeat the cycle.

Both approaches are closely related as they solve problem (3) $N_{samp}$ steps in advance using the predicted states at $k+N_{samp}$ from the current optimal solution. Moreover, as shown in Yang and Biegler (2013), both satisfy nominal stability properties under mild assumptions. On the other hand, the serial approach uses only one processor, where (3) is solved every $N_{samp}$ sampling times, and the first $N_{samp}$ control variables in the horizon are updated through an augmented sensitivity approach. The parallel approach uses multiple processors and problem (3) is initiated at every sampling time, with the first control variable updated using (9), but different KKT matrices are used at every sampling time. With either approach, amsNMPC generally reduces on-line computational cost by two to three orders of magnitude, as with the asNMPC approach. We also note that amsNMPC with $N_{samp}=0$, $N_{samp}=1$ and $N_{samp}=\infty$ corresponds to ideal NMPC, asNMPC and the basic NEU approach (Würth et al. (2009)), respectively.

### *6.1 Distillation column case study*

To demonstrate the amsNMPC approach, we consider a large-scale propane-propylene distillation column. The distillation model has the same DAE structure as in Section 4.1 but now with 158 trays, and equilibrium constants approximated from DePriester nomographs. The controlled states are the concentrations of propylene of the first and last trays, while the control variables are the steam pressure in the reboiler and the bottoms flow rate. The objective is to keep the states at their setpoints. This case study compares the performance of ideal NMPC and the parallel approach of amsNMPC. After discretizing the ordinary differential equations with orthogonal collocation, the NLP problem has 111650 variables and 111580 constraints. Solution of problem (3) requires 90 CPU seconds, but with a sampling time of only 60 s, $N_{samp}$ must be greater than one, and amsNMPC must be used. To show the difference in performance with different $N_{samp}$; we assume we can choose $N_{samp}=0$ (ideal NMPC), $N_{samp}=1$ (asNMPC) as well as $N_{samp}=2$ and $N_{samp}=3$.

For this case, we change the setpoint at $t=30$ and introduce Gaussian noise with a standard deviation of 0.05% on the output measurements. Fig. 3 shows that there is not much difference among different cases in the states profile. Increasing the noise level to 1% on all outputs but $x[Ntray]$ in Fig. 4, shows that the difference in the state profiles becomes larger as noise level increases. With $N_{samp}=3$, the deviation from the performance of ideal NMPC is the largest. As expected, NMPC performance generally degrades as $N_{samp}$ increases, especially for nonlinear systems with noise. Additional performance results and analysis can be found in Yang and Biegler (2013).

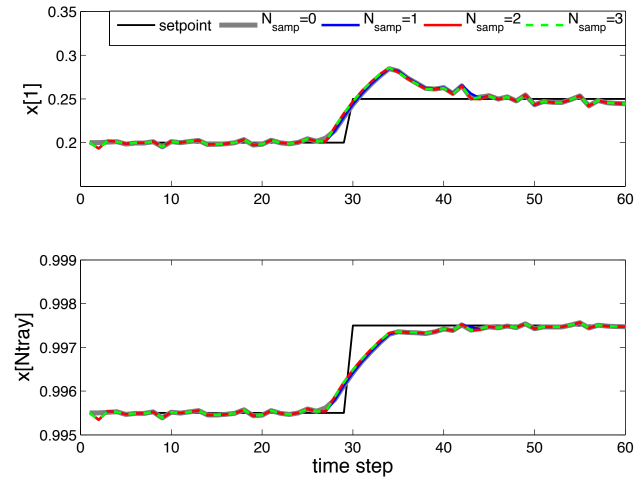


Figure 3. Performance of the parallel amsNMPC with 0.05% measurement noise

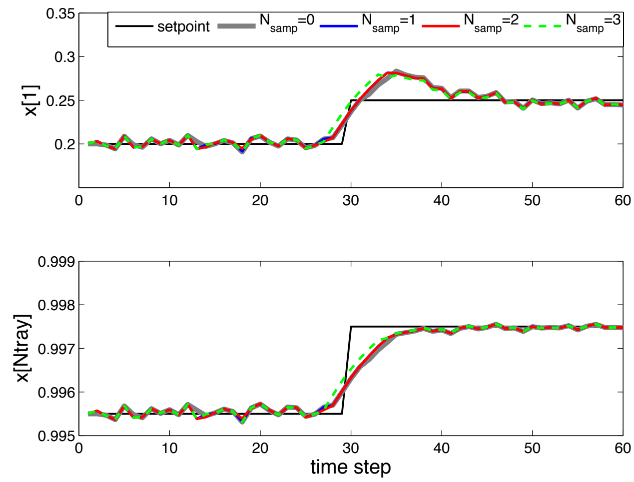


Figure 4. Performance of the parallel amsNMPC with higher level of noise

## 7. ECONOMIC NMPC AND DYNAMIC REAL-TIME OPTIMIZATION

As explored in Biegler and Zavala (2009), combining the two-layer RTO approach into a single dynamic optimization may lead to significant improvements in process performance. In particular, *artificial setpoints* used in (3) and determined from a steady state optimization are no longer needed. Instead, a dynamic optimization directly maximizes an economic objective using a well-tuned dynamic process model.

However, economic NMPC or, equivalently, dynamic real-time optimization, is not as simple as replacing the stage costs in (3); stability and robustness properties are still required. Moreover, since the setpoint, i.e., the 'zero' in Theorem 1, is unknown, we need to consider how stability can be guaranteed.

The most direct way to enforce stability of economic NMPC is to consider an infinite time horizon, $N\to\infty$. Significant early advances have been made in designing NMPC controllers along these lines (Angeli et al. (2012); Würth et al. (2009)), although many issues relating to computation and robust stablity remain. A more desirable result would be to assume that Economic NMPC drives the process to a steady state. However, this assumption cannot be guaranteed and counterexamples have been presented in Angeli et al. (2012).

Nevertheless, there are several reasons to require convergence to a steady state. In particular, production planning with economic models over long time scales is based on steady state models, and consistency with these models must be ensured. Also, plant stability and robustness are easier to analyze under steady state assumptions. Finally, steady state (or cyclic steady state) operation is easier to monitor, analyze and manage.

To ensure that Economic NMPC converges to steady state we revisit NMPC stability analysis through the following constructions. We first define the optimal steady state. To establish the Lyapunov function, a transformed system is introduced by subtracting the optimal steady state from the original system. Here, the original system is asymptotically stable at the optimum if the transformed system is asymptotically stable at the origin. In addition, we need to show that the transformed Lyapunov function is strictly decreasing. This requires an additional modification to create a *rotated* Lyapunov function, related to the original system.

To define implicit reference values for the states and controls, we consider the steady state optimization problem given by:

$$\min \psi(z,v), \text{ s.t. } z = f(z,v), z \in \mathbb{X}, v \in \mathbb{U} \tag{20}$$

with the solution given by $(z^*, v^*)$. We introduce a transformed system by subtracting the optimal steady state from the predicted values as follows:

$$\bar{z}_l = z_l - z^*, \quad \bar{v}_l = v_l - v^* \tag{21}$$

and the transformed state evolves according to

$$\bar{z}_{l+1} = \bar{f}(\bar{z}_l, \bar{v}_l) = f(\bar{z}_l + z^*, \bar{v}_l + v^*) - z^* \tag{22}$$

and $\bar{z}_l \in \bar{\mathbb{X}}$ and $\bar{u}_l \in \bar{\mathbb{U}}$, where $\bar{\mathbb{X}}$ and $\bar{\mathbb{U}}$ are the corresponding sets for the transformed system. From equation (22), we see that when $(\bar{z}_l, \bar{v}_l) = (0,0)$,

$$z^*_{l+1} = f(z^*_l, v^*_l) = z^* \tag{23}$$

Similarly, we define the transformed stage and terminal costs as:

$$\bar{\psi}(\bar{z}_l, \bar{v}_l) = \psi(\bar{z}_l + z^*, \bar{v}_l + v^*) - \psi(z^*, v^*) \tag{24}$$

$$\bar{\Psi}(\bar{z}_l, \bar{v}_l) = \Psi(\bar{z}_l + z^*, \bar{v}_l + v^*) - \Psi(z^*, v^*) \tag{25}$$

$$\tag{26}$$

and $\bar{\Psi}(0,0) = \bar{\psi}(0,0) = 0$, so that the NMPC subproblem is now given by:

$$\min \quad \bar{\Psi}(\bar{z}_N) + \sum_{l=0}^{N} \bar{\psi}(\bar{z}_l, \bar{v}_l) \tag{27}$$

$$\text{s.t.} \quad \bar{z}_{l+1} = \bar{f}(\bar{z}_l, \bar{v}_l), \quad l = 0, \ldots, N$$

$$\bar{z}_0 = \tilde{x}_k - x^*_k,\ \bar{z}_l \in \bar{\mathbb{X}},\ \bar{v}_l \in \bar{\mathbb{U}},\ \bar{z}_N \in \bar{\mathbb{X}}_f,$$

A key concern is that Assumption 1(4) generally does not hold for transformed economic stage costs $\bar{\psi}$ and does not directly lead to a Lyapunov function. Instead, Diehl et al. (2011) suggested the alternative *rotated* stage cost given by:

$$L(\bar{z}_l, \bar{v}_l) = \bar{\psi}(\bar{z}_l, \bar{v}_l) + \bar{\lambda}^T(\bar{z}_l - \bar{f}(\bar{z}_l, \bar{v}_l)). \tag{28}$$

where $\bar{\lambda}$ is the multiplier from the equality constraints in (20). This function has a local minimum at $(0,0)$ and a global minimum if $L$ is convex. Moreover, this stage cost satifies Assumption 1(4) if $L$ is strongly convex. From $\bar{f}_i(0,0) = 0$, $L(\bar{z}, \bar{v}) \geq 0$, $L(0,0) = 0$, and $[\bar{z}^T \bar{v}^T]\nabla L(0,0) \geq 0$ we can write:

$$\begin{aligned} L(\bar{z}, \bar{v}) &= \nabla L(0,0)^T \begin{bmatrix} \bar{z} \\ \bar{v} \end{bmatrix} + 1/2 \int_0^1 [\bar{z}^T\ \bar{v}^T] \nabla^2 L(\tau\bar{z}, \tau\bar{v}) \begin{bmatrix} \bar{z} \\ \bar{v} \end{bmatrix} d\tau \\ &\geq 1/2 \int_0^1 [\bar{z}^T\ \bar{v}^T] \nabla^2 L(\tau\bar{z}, \tau\bar{v}) \begin{bmatrix} \bar{z} \\ \bar{v} \end{bmatrix} d\tau. \end{aligned} \tag{29}$$

If $L(\bar{z}_i, \bar{v}_i)$ is strongly convex, we have for some $\gamma_L > 0$,

$$\begin{aligned} L(\bar{z}, \bar{v}) &\geq 1/2 \int_0^1 [\bar{z}^T\ \bar{v}^T] \nabla^2 L(\tau\bar{z}, \tau\bar{v}) \begin{bmatrix} \bar{z} \\ \bar{v} \end{bmatrix} d\tau \\ &\geq \gamma_L(|\bar{z}|^2 + |\bar{v}|^2) \geq \beta_\infty(|\bar{z}|) \end{aligned} \tag{30}$$

and Assumption 1(4) holds. If (30) does not hold, regularization terms may be added to the *original* stage cost in (3) for economic NMPC.

$$\psi(z_l, v_l) + \|z_l - z^*\|^2_Q + \|v_l - v^*\|^2_R \tag{31}$$

where $Q, R$ are suitably defined weighting matrices. Such a regularization leads to the desired convexity property for the rotated stage costs and ensures that Assumption 1(4) holds. In addition, the terminal cost can also be considered by defining the *rotated terminal cost*: $L_F(z) = \bar{\Psi}(\bar{z}) + \bar{\lambda}^T \bar{z}$.

Huang et al. (2012a) analyze the case of regularized Economic NMPC formulations and prove nominal and ISS stability properties. Moreover, they extended this formulation to dynamic processes that operate in a cycle steady state over a period of $K$ sampling times. (The particular case with $K = 1$ is the normal steady state). Such systems arise under periodic variation of input conditions such as feedstocks, prices and boundary conditions. These processes, which are never in steady state but operate in cycles, include periodic adsorption processes, reactors with reaction/regeneration cycles and processes which follow diurnal variations in energy prices.

Within this optimization framework, Economic NMPC has been demonstrated on a number of process applications. Amrit et al. (2013) consider two literature processes and an extensive set of case studies compare performance and costs between tracking-type and Economic NMPC. These studies show that optimizing process economics takes advantage of the type and frequency of disturbances and can lead to an economic advantage of 5-10% in steady-state profit over tracking-type controllers.

For the dynamic real-time optimization of low density polyethylene (LDPE) reactors, significant production improvements can be made during fouling and defouling cycles (Zavala and Biegler (2009b)). In particular, optimal adjustment of the temperature profile can increase production levels by more than 10% compared to setpoint tracking.

For the control of compressors in gas pipeline applications, a number of advantages can be seen in tracking diurnal variations in electricity prices and cost savings of 5% were observed when choosing real-time pricing over flat-rate schemes (Gopalakrishnan and Biegler (2013)).

Finally, for thermo-mechanical pulping in the paper industry, the integration of economic objectives with an NMPC formulation leads to potential reductions in energy cost of about 12% for current process configurations (Harinath et al. (2011)), and up to 24% with the addition of multiple pulping stages (Harinath et al. (2013)).

### *7.1 Economic NMPC for Air Separation*

To demonstrate Economic NMPC with a periodic cycle of $K$ sampling times, Huang et al. (2012a) consider the dynamic model of an air separation unit. The unit contains two integrated cryogenic distillation columns, each with 40 trays. The high pressure column operates at 5-6 bars, while the low pressure column operates at 1-1.5 bars and also has 40 trays. An air feed flow is split into two substreams. The high pressure air

(MA) enters the bottom of the high pressure column and the expanded air (EA) enters the $20^{th}$ tray of the low pressure column. Crude nitrogen gas (GN) from the main heat exchanger is also added to the $25^{th}$ tray of the high pressure column. The reboiler of the low pressure column is integrated with the condenser of the high pressure column. The main products of the high pressure column are pure nitrogen (PNI) ($> 99.99\%$) and crude liquid oxygen ($\sim 50\%$). The low pressure column has nitrogen product with $\sim 99\%$ purity and oxygen product (POX) with $\sim 97\%$ purity. The ASU model is represented by tray-by-tray MESH equations for nitrogen, oxygen and argon and can be found in Huang et al. (2009). The model is composed of 320 differential equations and 1,200 algebraic equations. Upon discretization, problem (3) has 117,140 variables and 116,900 constraints.

The control structure is reported in Huang et al. (2009). We choose the molar flow rate of product oxygen (POX-$Y_1$), product nitrogen (PNI-$Y_2$), the temperature at $30^{th}$ tray in the low pressure column (Tl30-$Y_3$), and temperature at the $15^{th}$ tray in the high pressure column (Th15-$Y_4$) as output variables. Four stream flow rates are considered as control variables, including the expanded air feed (EA-$U_1$), main air feed (MA-$U_2$), reflux liquid nitrogen (LN-$U_3$) and crude nitrogen gas (GN-$U_4$). The stage cost is the electricity cost for the air feed compressor, which equals $(MA + EA) \times$ electricity price. To demonstrate periodic operation, a sinusoidally varying electricity price with period of 60 minutes is used.

The NMPC formulation (20) with a periodic constraint is also used to control the system. To guarantee the stability of the closed loop system, a regularization term (31) is added to the stage cost, with reference values for the outputs. Closed loop responses are shown in Fig. 5 and 6. It is easy to see that both input flowrates (EA and MA) exhibit sinusoidal behavior, and are at their minimum when the electricity price is the highest. Depending on the electricity pricing structure, Huang et al. (2012a) report a 4-6% reduction in energy costs over optimal setpoint tracking.

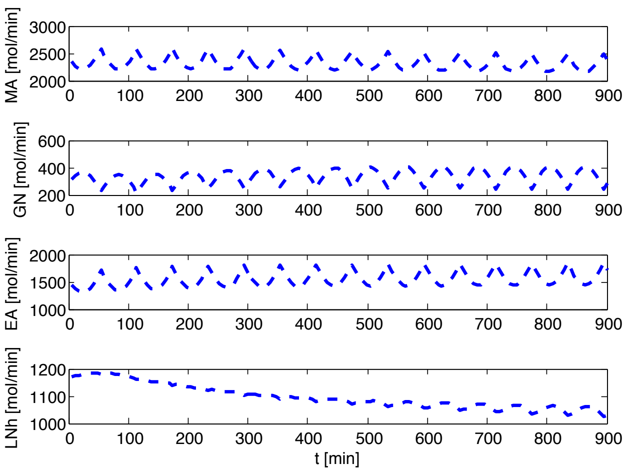


Figure 5. Input profile for periodic ASU with Economic NMPC

## 8. CONCLUSIONS AND FUTURE WORK

Advances in large-scale nonlinear programming solvers and sensitivity allow the formulation of nonlinear model-based estimation (MHE) and control (NMPC) that require only negligible on-line computation. This allows us to expand the scope of control activities from mere setpoint regulation to trajectory tracking and dynamic real-time optimization. Moreover, it allows us to incorporate large-scale nonlinear process models into dynamic on-line optimization strategies.

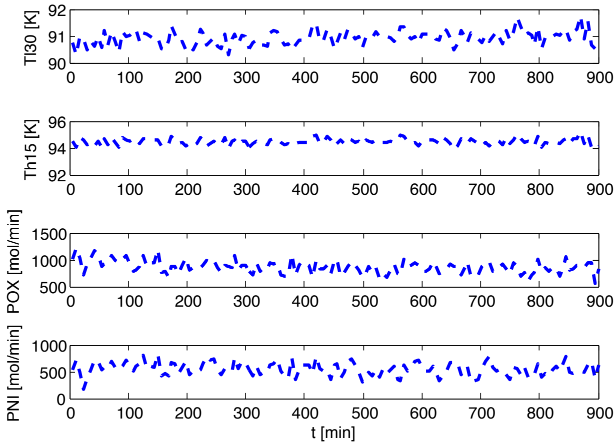


Figure 6. Output profile for periodic ASU with Economic NMPC

This study reviews recent results related to *advanced step* MHE and NMPC. We summarize nominal and robust stability properties for advanced step NMPC and discuss the extension of this strategy to background calculations that are performed over multiple sampling times. For MHE we describe a fast method for the covariance update of arrival costs and we modify the MHE formulation through the incorporation of M-estimators, in order to remove the estimation bias due to gross measurement errors. Finally, we extend the NMPC formulation to consider economic stage costs that lead directly to dynamic real-time optimization. All of these advances are illustrated with distillation case studies with large-scale first principle models.

These advances also lead to a number of open questions. Related to robust stability of NMPC, the ISS property assumes a robust positive invariant (RPI) set but does not specify a guaranteed uncertainty margin. Future analysis with direct calculation of RPI regions, possibly with tube-based NMPC approaches need to be considered (Raković et al. (2006)). Along with amsNMPC, multi-step variants of MHE also will be developed along with deeper analysis of robust stability that extends the results of Theorem 2. In addition, the use of M-estimators leads to a promising strategy for detection of failed sensors and process faults; it also extends the distributional assumptions of the MHE formulation. Finally, recent results for Economic NMPC raise interesting questions on the realization of dynamic RTO, as well as the opportunity costs of operating at steady state and the degree of regularization required.

## ACKNOWLEDGMENTS

Many thanks to my research colleagues Eranda Harinath, Rui Huang, Rodrigo Lopez Negrete, Bethany Nicholson, Sachin Patwardhan, Victor Zavala, and Xue Yang for very useful input and discussions. Funding from National Science Foundation, Department of Energy (NETL) and Center for Advanced Process Decision-Making (CAPD) at Carnegie Mellon is gratefully acknowledged.

REFERENCES

Abrol, S. and Edgar, T.F. (2011). A fast and versatile technique for constrained state estimation. *Journal of Process Control*, 21(3), 343 – 350.

Adetola, V. and Guay, M. (2010). Integration of real-time optimization and model predictive control. *Journal of Process Control*, 20, 125–133.

Alamir, M. (2000). A framework for monitoring control updating period in real-time NMPC schemes. In F.A.E. L. Magni D. Raimondo (ed.), *Nonlinear Model Predictive Control*, volume 384 of *Lecture Notes in Control and Information Sciences*, 433–445. Springer, Berlin.

Alessandri, A., Baglietto, M., and Battistelli, G. (2008). Moving-horizon state estimation for nonlinear discrete-time systems: New stability results and approximation schemes. *Automatica*, 44(7), 1753–1765.

Amrit, R., Rawlings, J.B., and Biegler, L.T. (2013). Optimizing process economics online using model predictive control. *Comp. Chem. Engr.*, to appear.

Angeli, D., Amrit, R., and Rawlings, J. (2012). On average performance and stability of economic model predictive control. *IEEE Trans. Auto. Cont.*, 57 (7), 1615–1626.

Arora, N. and Biegler, L.T. (2001). Redescending estimators for data reconciliation and parameter estimation. *Computers & Chemical Engineering*, 25(1112), 1585 – 1599.

Arulampalam, S., Maskell, S., Gordon, N., and Clapp, T. (2002). A tutorial on particle filters for on-line nonlinear/non-gaussian bayesian tracking. *IEEE Transactions on Signal Processing*, 50, 174–188.

Aske, E., Strand, S., and Skogestad, S. (2008). Coordinator mpc for maxmizing plant throughput. *Computers & Chemical Engineering*, 32, 195–204f.

Biegler, L.T. and Zavala, V.M. (2009). Large-scale nonlinear programming using ipopt: An integrating framework for enterprise-wide dynamic optimization. *Computers and Chemical Engineering*, 33, 575–582.

Bryson, A.E. and Ho, Y.C. (1975). *Applied Optimal Control: Optimization, Estimation, and Control*. Taylor and Francis, New York, USA.

Chen, H. and Allgöwer, F. (1998). A quasi-infinite horizon nonlinear model predictive control scheme with guaranteed stability. *Automatica*, 34, 1205–1217.

Daum, F. (2005). Nonlinear filters: beyond the kalman filter. *Aerospace and Electronic Systems Magazine, IEEE*, 20(8), 57–69.

Diehl, M., Amrit, R., and Rawlings, J.B. (2011). A lyapunov function for economic optimizing model predictive control. *IEEE Trans. Auto. Cont*, 56(3), 703–707.

Diehl, M., Bock, H.G., and Schlöder, J.P. (2005a). A real-time iteration scheme for nonlinear optimization in optimal feedback control. *SIAM J. Control and Optimization*, 43, 1714–1736.

Diehl, M., Bock, H., and Schlöder, J. (2005b). A real-time iteration scheme for nonlinear optimization in optimal feedback control. *SIAM Journal on Control and Optimization*, 43, 1714–1736.

Diehl, M., Uslu, I., Findeisen, R., Schwarzkopf, S., Allgöwer, F., and et al (2002). Real-time optimization for large scale processes: Nonlinear model predictive control of a high purity distillation column. *Journal of Process Control*, 12, 577–585.

Duff, I. (2004). Ma57 - a code for the solution of sparse symmetric definite and indefinite systems. *ACM Transactions on Mathematical Software*, 30, 118–144.

Engell, S. (2007). Feedback control for optimal process operation. *J. Proc. Cont.*, 17, 203–219.

Evensen, G. (1994). Sequential data assimilation with a nonlinear quasi-geostrophic model using monte carlo methods to forecast error statistics. *Journal of Geophisical Research*, 99(C5), 143–162.

Fiacco, A. (1983). *Introduction to Sensitivity and Stability Analysis in Nonlinear Programming*. Academic Press, New York.

Findeisen, R. and Allgöwer, F. (2004). Computational delay in nonlinear model predictive control. In *Proc. Int. Symp. Adv. Control of Chemical Processes*, 427–432. Hong Kong.

Findeisen, R., Imsland, L., Allgöwer, F., and Foss, B. (2003). State and output feedback nonlinear model predictive control: an overview. 9, 190–206.

Forsgren, A., Gill, P., and Wright, M. (2002). Interior methods for nonlinear optimization. *SIAM Review*, 44/4, 525–597.

Gopalakrishnan, A. and Biegler, L. (2013). Economic nonlinear model predictive control for the periodic optimal operation of gas pipeline networks. *Computers and Chemical Engineering*, 52, 90–99.

Hampel, F.R. (1974). The influence curve and its role in robust estimation. *Journal of the American Statistical Association*, 69(346), 383–393.

Harinath, E., Biegler, L.T., and Dumont, G.A. (2011). Control and optimization strategies for thermo-mechanical pulping processes: Nonlinear model predictive control. *Journal of Process Control*, 21, 519 – 528.

Harinath, E., Biegler, L., and Dumont, G.A. (2013). Predictive optimal control for thermo-mechanical pulping processes with multi-stage low consistency refining. *Journal of Process Control*, 23, 1001 – 1011.

Haseltine, E.L. and Rawlings, J.B. (2005). Critical evaluation of extended kalman filtering and moving-horizon estimation. *Ind. Eng. Chem. Res.*, 44, 2451–2460.

Huang, R., Biegler, L., and Patwardhan, S. (2010a). Fast offset-free nonlinear model predictive control based on moving horizon estimation. *Ind. Eng. Cheme. Res*, 49, 7882–7890.

Huang, R., Biegler, L.T., and Harinath, E. (2012a). Robust stability of economically oriented infinite horizon NMPC that include cyclic processes. *J. Process Control*, 22 (1), 51–59.

Huang, R., Patwardhan, S., and Biegler, L. (2010b). Stability of a class of discrete-time nonlinear recursive observers. *J. of Process Control*, 20, 1150–1160.

Huang, R., Patwardhan, S.C., and Biegler, L.T. (2012b). Robust stability of nonlinear model predictive control based on extended kalman filter. *Journal of Process Control*, 22, 82–89.

Huang, R., Zavala, V., and Biegler, L. (2009). Advanced step nonlinear model predictive control for air separation units. *Journal of Process Control*, 19, 678–685.

Huber, P.J. (1981). *Robust Statistics*. John Wiley and Sons, New York.

Jazwinski, A.H. (2007). *Stochastic Processess and Filtering Theory*. Dover Publications, Inc., Mineola, New York.

Julier, S., Uhlmann, J., and Durrant-Whyte, H. (2000). A new method for the nonlinear transformation of means and covariances in filters and estimators. *IEEE Transactions on Automatic Control*, 45(3), 477–482.

Kuehl, P., Diehl, M., Kraus, T., Schloeder, J.P., and Bock, H.G (2011). A real-time algorithm for moving horizon state and parameter estimation. *Computers & Chemical Engineering*, 35(1), 71 – 83.

Li, W. and Biegler, L.T. (1989). Multistep, newton-type control strategies for constrained, nonlinear processes. *Chem. Eng. Res. Des.*, 67, 562–577.

López-Negrete, R. (2011). *Nonlinear Programming Sensitivity Based Methods for Constrained State Estimation*. Ph.D. thesis, Carnegie Mellon University, Pittsburgh, PA, USA.

López-Negrete, R. and Biegler, L.T. (2012). A moving horizon estimator for processes with multi-rate measurements: A nonlinear programming sensitivity approach. *Journal of Process Control*, 22(4), 677 – 688.

López-Negrete, R., DAmato, F.J., Biegler, L.T., and Kumar, A. (2013). Fast nonlinear model predictive control: Formulation and industrial process applications. *Computers & Chemical Engineering*, 51, 55 – 64.

Magni, L. and Scattolini, R. (2007). Robustness and robut design of mpc for nonlienar descrete-time systems. In R. Findeisen, F. Allgöwer, and L. Biegler (eds.), *Assessment and Future Directions of Nonlinear Model Predictive Control*, 239–254. Springer, Berlin.

Mayne, D., Rawlings, J., Rao, C., and Scokaert, P. (2000). Constrained model predictive control: stability and optimality. *Automatica*, 36, 789–814.

Michalska, H. and Mayne, D. (1995). Moving horizon observers and observer-based control. *Automatic Control, IEEE Transactions on*, 40(6), 995–1006.

Muske, K. and Rawlings, J. (1993). Receding horizon recursive estimation. In *Proceedings of the American Control Conference, June, San Fransisco, CA*.

Nicholson, B.L., Lopez-Negrete, R., and Biegler, L.T. (2013). On-line state estimation of nonlinear dynamic systems with gross errors. *submitted for publication*.

Ohtsuka, T. (2004). A continuation/GMRES method for fast computation of nonlinear receding horizon control. *Automatica*, 40 (4), 563 – 574.

Ohtsuka, T. and Fujii, H. (1996). Nonlinear receding-horizon state estimation by real-time optimixation technique. *Journal of Guidance, Control, and Dynamics*, 19(4), 863–870.

Pirnay, H., López-Negrete, R., and Biegler, L.T. (2012). Optimal sensitivity based on ipopt. *Math. Programming Computation*, 4, 307 – 331.

Prakash, J., Patwardhan, S.C., and Shah, S.L. (2010). Constrained nonlinear state estimation using ensemble kalman filters. *Industrial & Engineering Chemistry Research*, 49(5), 2242–2253.

Prakash, J., Shah, S., and Patwardhan, S. (2008). Constrained state estimation using particle filters. In M.J. Chung and P. Misra (eds.), *Proceedings of the 17th IFAC World Congress*, volume 17, 6472–6477.

Raković, S., Teel, A., Mayne, D., and Astolfi, A. (2006). Simple robust control invariant tubes for some classes of nonlinear discrete time systems. In *Proceedings of the 45th IEEE Conference on Decision & Control*, 6397–6402. San Diego, CA, USA.

Rao, C.V., Rawlings, J.B., and Mayne, D.Q. (2003). Constrained state estimation for nonlinear discrete-time systems: Stability and moving horizon approximations. *IEEE Trans. Auto. Cont.*, 48(2), 246–258.

Rawlings, J.B. and Mayne, D.Q. (2009). *Model Predictive Control: Theory and Design*. Nob Hill Publishing, LLC.

Rawlings, J. and Amrit, R. (2009). Optimizing process economic performance using model predictive control. In L. Magni, D.M. Raimondo, and F. Allgöwer (eds.), *Assessment and Future Directions of Nonlinear Model Predictive Control*, 119–138. Springer.

Rawlings, J., Bonné, D., Jørgensen, J., Venkat, A., and Jørgensen, S. (2008). Unreachable setpoint in model predictive control. *IEEE Trans. on Auto. Cont.*, 53, 2209–2215.

Robertson, D.G., Lee, J.H., and Rawlings, J.B. (1996). A moving horizon-based approach for least-squares estimation. *AIChE Journal*, 42(8), 2209–2224.

Tenny, M. and Rawlings, J. (2002). Efficient moving horizon estimation and nonlinear model predictive control. In *Proceedings of the American control conference, Anchorage, AK*.

Vachhani, P., Narasimhan, S., and Rengaswamy, R. (2006). Robust and Reliable Estimation via Unscented Recursive Nonlinear Dynamic Data Reconciliation. *J. Process Control*, 16, 1075–1086.

Vachhani, P., Rengaswamy, R., Gangwal, V., and Narasimhan, S. (2004). Recursive estimation in constrained nonlinear dynamical systems. *AIChE Journal*, 51(3), 946–959.

Wolf, I., Würth, L., and Marquardt, W. (2011). Rigorous solution vs. fast update: Acceptable computational delay in NMPC. In *Decision and Control and European Control Conference, Orlando, Florida, USA*, 5230–5235.

Würth, L., Hannemann, R., and Marquardt, W. (2009). Neighboring-extremal updates for nonlinear model-predictive control and dynamic real-time optimization. *J. Process Control*, 19, 1277–1288.

Würth, L., Rawlings, J.B., and Marquardt, W. (2009). Economic dynamic real-time optimization and nonlinear model predictive control on infinite horizons. In *International Symposium on Advanced Control of Chemical Process*. Istanbul, Turkey.

Yang, X. and Biegler, L.T. (2013). Advanced-multi-step nonlinear model predictive control. *J. Process Control*, to appear.

Zavala, V.M. and Biegler, L.T. (2000). Nonlinear programming strategies for state estimation and model predictive control. In F.A.E. L. Magni D. Raimondo (ed.), *Nonlinear Model Predictive Control*, volume 384 of *Lecture Notes in Control and Information Sciences*, 419–432. Springer, Berlin.

Zavala, V.M. and Biegler, L.T. (2009a). The advanced-step NMPC controller: optimality, stability and robustness. *Automatica*, 45, 86–93.

Zavala, V.M. and Biegler, L. (2009b). Optimization-based strategies for the operation of low-density polyethylene tubular reactors: Nonlinear model predictive control. *Computers and Chemical Engineering*, 33, 1735–1746.

Zavala, V.M., Laird, C.D., and Biegler, L.T. (2008). A fast moving horizon estimation algorithm based on nonlinear programming sensitivity. *Journal of Process Control*, 18(9), 876–884.