A Fast Moving Horizon Estimation Algorithm Based on Nonlinear Programming Sensitivity

Victor M. Zavala*, Carl D. Laird and Lorenz T. Biegler

Department of Chemical Engineering, Carnegie Mellon University, 5000 Forbes Avenue, Pittsburgh PA, 15213, USA

Abstract

Moving Horizon Estimation (MHE) is an efficient optimization-based strategy for state estimation. Despite the attractiveness of this method, its application in industrial settings has been rather limited. This has been mainly due to the difficulty to solve, in real-time, the associated dynamic optimization problems. In this work, a fast MHE algorithm able to overcome this bottleneck is proposed. The strategy exploits recent advances in nonlinear programming algorithms and sensitivity concepts. A detailed analysis of the optimality conditions of MHE problems is presented. As a result, strategies for fast covariance information extraction from general nonlinear programming algorithms are derived. It is shown that highly accurate state estimates can be obtained in large-scale MHE applications with negligible on-line computational costs.

Keywords: estimation algorithms, real-time, large-scale, nonlinear programming, sensitivity analysis.

1 Introduction

Moving horizon estimation has been identified as an efficient method for state estimation for constrained, linear and nonlinear systems. From a theoretical point of view, a deeper understanding of the estimator filtering and stability properties has led to efficient formulations with guaranteed stability properties [1, 2, 3, 4]. In addition to this, increased interest in MHE has resulted from its proven superiority over traditional estimation approaches such as the extended Kalman filter (EKF) [5]. The main practical advantages of MHE is that it allows to handle complex nonlinear dynamic models directly and to incorporate constraints. The main limitation of MHE is that it requires on-line solutions of dynamic optimization problems. This limitation becomes particularly relevant in large-scale industrial applications where the solution of the MHE problem takes a non-negligible amount of time, giving rise to computational delays.

The deterioration of performance and stability of NMPC strategies due to computational delays has been studied in [6, 7]. As industry demands the implementation of increasingly larger applications,

*Author to whom correspondence should be addressed. E-mail: vzavala@andrew.cmu.edu, Phone: (412) 268-2238

the development of strategies able to overcome these limitations becomes essential. In the context of NMPC, recently established connections between the parametric properties of NMPC, Newton-based convergence properties and the conceptual separation of background-feedback computations have dramatically changed the perspective on how to solve on-line NMPC problems efficiently [8, 9, 10]. Many different variants of fast or real-time NMPC strategies have been proposed recently. Alternatives include Newton-step controllers, neighboring extremals and NLP sensitivity-based NMPC controllers. Here, the main idea is to provide instantaneous approximate feedback to the plant based on a frequently updated reference problem solved in background. A related Newton-step MHE estimator has been recently proposed in [11]. Here, the estimator performs a single iteration (full Newton step) in the solution of the nonlinear MHE problem at each time step. This allows a fast estimation mechanism that has shown good practical performance. They show that if the series of Newton steps is initialized around a sufficiently good reference solution, then the full Newton steps can converge to the solution of the MHE problems. This result follows from the local convergence properties of Newton's method and the parametric properties of the MHE problem.

Newton-step strategies are built on the assumption that real-time iterations stay *sufficiently* close to the reference solution where local convergence can be guaranteed [12]. This assumption is, in general, unnecessarily restrictive and might not hold true in highly nonlinear or ill-posed problems, where non-convexity effects need to be handled adequately. Motivated by these observations, a fast MHE estimator based on background optimizations and NLP sensitivity concepts to obtain fast on-line approximations has been proposed in [13]. In this work, we extend these results with a deeper and more concise analysis of computational and theoretical issues associated to this strategy. The fast MHE estimator can be seen as the counterpart of the recently proposed advanced-step NMPC controller [10]. The next section presents the MHE problem under consideration. Here, we analyze the optimality conditions and general properties of MHE problems. In Section 3 we derive computationally efficient strategies for NLP sensitivity and extraction of covariance information from general NLP solvers. In Section 4, we make use of these results to propose a NLP-sensitivity based shifting strategy for fast MHE. The algorithmic framework is then applied to two simulated case studies and the results are presented in Section 5 while the last section concludes the paper.

2 MHE Problem

2.1 Conceptual Formulation

Consider the scenario in which a given system is located at sampling time t_k and a past measurement sequence $\{y(k-N), \ldots, y(k-1), y(k)\}$ is available. In this work, we consider that the MHE estimator

uses the *perfect* nonlinear model,

$$\begin{aligned} z_{l+1|k} &= f(z_{l|k}, w_{l|k}),\ l = 0, .., N-1 \\ y_{0|k} &= h(z_k) \end{aligned}$$

to find the disturbance sequence $\{w_{0|k}, ..., w_{N-1|k}\}$ and initial state $z_{0|k}$ that minimizes the cost function,

$$\phi(\eta(k)) = \Gamma(z_{0|k}) + L_N(z_N) + \sum_{l=0}^{N-1} L_l(z_l, w_l). \tag{1}$$

with,

$$\begin{aligned} \Gamma(z_{0|k}) &= (z_{0|k} - \bar{z}_0(k))^T \bar{\Pi}_0^{-1}(k)(z_{0|k} - \bar{z}_0(k)) \\ L_l(z_l, w_l) &= (y(k-N+l) - h(z_l))^T R_l^{-1}(y(k-N+l) - h(z_l)) + w_l^T Q_l^{-1} w_l, \qquad l = 0, ..., N-1 \\ L_N(z_N) &= (y(k) - h(z_N))^T R_N^{-1}(y(k) - h(z_N)) \end{aligned} \tag{2}$$

over a past horizon containing N time steps. Here, the computed disturbances $w_{0|k} \in \Re^{nw}$ and states $z_{0|k} \in \Re^{nz}$ are enforced to satisfy the constraints $w_{l|k} \in \mathbb{W}$ $z_{l|k} \in \mathbb{Z}$, respectively, $\forall\, l, k$. The cost function $\phi : \Re^{Nnz+(N-1)nw} \to \Re$ comprises a set of stage costs $L_l : \Re^{nx+nu} \to \Re$ and of an initial penalty term $\Gamma : \Re^{nz} \to \Re$ summarizing prior information before t_{k-N}. Here $\bar{z}_0(k)$ is the *a priori* state estimate with associated covariance $\bar{\Pi}_0(k)$. The matrices $\bar{\Pi}_0(k)$, Q_l, and R_l are assumed to be symmetric positive definite. In addition, we define the problem data,

$$\eta(k) = (\bar{z}_0(k), \bar{\Pi}_0^{-1}(k), y(k-N), ..., y(k-1), y(k)) \tag{3}$$

that fully defines the current MHE problem. In compact form, this leads to a nonlinear programming problem of the form,

$$\begin{aligned} \mathcal{M}_N(\eta(k)) \quad \min_{z_{0|k}, w_{l|k}} \ \phi(\eta(k)) &= \Gamma(z_{0|k}) + L_N(z_N) + \sum_{l=0}^{N-1} L_l(z_l, w_l) \\ \text{s.t. } z_{l+1|k} &= f(z_{l|k}, w_{l|k}), \qquad l = 0, ..., N-1 \\ z_{l|k} &\in \mathbb{Z},\ w_{l|k} \in \mathbb{W}. \end{aligned} \tag{4}$$

From the solution of this problem we obtain the optimal disturbance sequence $\{w^*_{0|k} ... w^*_{N-1|k}\}$ and state sequence $\{z^*_{0|k} ... z^*_{N|k}\}$ from which we extract the current state estimate of the plant $x(k) = z^*_{N|k}$.

At the next sampling time t_{k+1}, we get the new measurement $y(k+1)$ and define the new problem data $\eta(k+1)$. For this, the measurements are shifted forward by one sampling time in order to drop the oldest measurement and include the current one, giving rise to the measurement sequence

$\{y(k-N+1),\ldots,y(k),y(k+1)\}$. In addition, the *a priori* state estimate is updated by defining $\bar{z}_0(k+1) = z^*_{1|k}$. Finally, the associated covariance matrix is updated. As a standard practice, an EKF update is used [3],

$$\bar{\Pi}_0(k+1) = G_0 Q_0 G_0^T + A_0 \bar{\Pi}_0(k) A_0^T - A_0 \bar{\Pi}_0(k) C_0^T (R_0 + C_0 \bar{\Pi}_0(k) C_0^T)^{-1} C_0 \bar{\Pi}_0(k) A_0^T \tag{5}$$

where $A_0 = \nabla_z f(z^*_{0|k}, w^*_{0|k})$, $G_0 = \nabla_w f(z^*_{0|k}, w^*_{0|k})$, and $C_0 = \nabla_z h(z^*_{0|k})$. It is important to note that $\bar{\Pi}_0(k+1)$ is only an *approximate* representation of the true covariance information obtained from solving the full MHE problem $\mathcal{M}_N(\eta(k))$.

Having the updated problem data $\eta(k+1)$, a new MHE problem $\mathcal{M}_N(\eta(k+1))$ is solved to estimate the new state of the plant $x(k+1)$. In the following, if we assume that the MHE problems can be solved *instantaneously* at each sampling time, we refer to the resulting MHE algorithm as *ideal* MHE.

3 Solution of MHE Problem

The representation of the NLP problem $\mathcal{M}_N(\cdot)$ is only conceptual since, in large-scale applications, the model $f(\cdot,\cdot)$ is developed implicitly from a set of continuous-time differential and algebraic equations (DAEs). Three main approaches are commonly used to solve MHE problems: single-shooting, multiple-shooting and simulatenous collocation. The main characteristics of MHE problems are the large number of degrees of freedom $(n_z + (n_w \times N))$ and common ill-posed formulations that lead to strong nonconvexities. Newton-based nonlinear programming algorithms using exact first and second order derivative information can handle these type of problems efficiently [14]. Exact derivative information can be obtained from modeling platforms or automatic differentiation routines.

3.1 Newton-Based NLP Algorithms

The MHE problem $\mathcal{M}(\cdot)$ is parametric in the data $\eta(\cdot)$ so we define the parameter $p = \eta(\cdot)$. In addition, we simplify the notation by dropping index k and adopt $z_{l|k} = z_l$, $w_{l|k} = w_l$, etc. The following results are based on the *post-optimal* analysis of solutions of parametric MHE problems. Consequently, we will simplify the presentation by handling the inequality constraints on the domains $\mathbb{Z}$ and $\mathbb{W}$ implicitly.

The Lagrange function associated to a nominal problem $\mathcal{M}(p_0)$ with $p = p_0$ is given by,

$$\mathcal{L} = \Gamma(z_0(p_0)) + L_N(z_N(p_0)) + \sum_{l=0}^{N-1} L_l(z_l(p_0), w_l(p_0)) + \sum_{l=0}^{N-1} \lambda_{l+1}(p_0)^T (z_{l+1}(p_0) - f(z_l(p_0), w_l(p_0))) \tag{6}$$

where $\lambda(p_0) \in \Re^{nz}$ are vectors of Lagrange multipliers. Note that all the primal variables and multipliers become implicit functions of p. To simplify the analysis, we suppress this dependence from the notation when the meaning is otherwise clear.

Any solution of a given MHE problem $\mathcal{M}(p_0)$ should satisfy the first-order Karush-Kuhn-Tucker (KKT) conditions,

$$\begin{aligned}
\nabla_{z_0}\mathcal{L} = \nabla_{z_0}\Gamma + \nabla_{z_0}L - \nabla_{z_0}f_0^T\lambda_1 &= 0 && && (7a)\\
\nabla_{z_l}\mathcal{L} = \nabla_{z_l}L_l + \lambda_l - \nabla_{z_l}f_l^T\lambda_{l+1} &= 0, && l = 1,...,N-1 && (7b)\\
\nabla_{w_l}\mathcal{L} = \nabla_{w_l}L_l - \nabla_{w_l}f_l^T\lambda_{l+1} &= 0, && l = 0,...,N-1 && (7c)\\
\nabla_{\lambda_{l+1}}\mathcal{L} = z_{l+1} - f_l &= 0, && l = 0,...,N-1 && (7d)\\
\nabla_{z_N}\mathcal{L} = \nabla_{z_N}L_N + \lambda_N &= 0 && && (7e)
\end{aligned}$$

where $f_l := f(z_l, w_l)$, $h_l := h(z_l)$, $L_l := L(z_l, w_l)$. For later reference we define $A_l = \nabla_{z_l} f_l$, $G_l = \nabla_{w_l} f_l$, $C_l = \nabla_{z_l} h_l$. Newton-Based NLP algorithms compute the search step toward the optimal solution by linearizing the nonlinear KKT conditions (7) around the current trajectory. This gives rise to the linear KKT system,

$$\begin{aligned}
P_0\Delta z_0 + F_0\Delta w_0 - A_0^T\Delta\lambda_1 &= -r_{z_0} && && (8a)\\
P_l\Delta z_l + F_l\Delta w_l + \Delta\lambda_l - A_l^T\Delta\lambda_{l+1} &= -r_{z_l} && l = 1,...,N-1 && (8b)\\
P_N\Delta z_N + \Delta\lambda_N &= -r_{z_N} && && (8c)\\
W_l\Delta w_l + F_l^T\Delta z_l - G_l^T\Delta\lambda_{l+1} &= -r_{w_l} && l = 0,...,N-1 && (8d)\\
\Delta z_{l+1} - A_l\Delta z_l - G_l\Delta w_l &= -r_{\lambda_{l+1}} && l = 0,...,N-1. && (8e)
\end{aligned}$$

If we make an explicit distinction between primal variables and multipliers the KKT system can be represented as,

$$\left[\begin{array}{cccccc|cccccc}
P_0 & F_0 & & & & & -A_0^T & & & & & \\
F_0^T & W_0 & & & & & -G_0^T & & & & & \\
 & & \ddots & & & & & \ddots & & & & \\
 & & & P_{N-1} & F_{N-1} & & & & \mathbb{I}_{n_z} & -A_{N-1}^T & & \\
 & & & F_{N-1}^T & W_{N-1} & & & & & -G_{N-1}^T & & \\
 & & & & & P_N & & & & & \mathbb{I}_{n_z} & \\
\hline
-A_0 & -G_0 & \mathbb{I}_{n_z} & & & & & & & & & \\
 & & \ddots & & & & & & & & & \\
 & & & -A_{N-1} & -G_{N-1} & \mathbb{I}_{n_z} & & & & & &
\end{array}\right]
\left[\begin{array}{c}
\Delta z_0\\ \Delta w_0\\ \vdots\\ \Delta z_{N-1}\\ \Delta w_{N-1}\\ \Delta z_N\\ \hline \vdots\\ \Delta\lambda_{N-1}\\ \Delta\lambda_N
\end{array}\right]
= -
\left[\begin{array}{c}
r_{z_0}\\ r_{w_0}\\ \vdots\\ r_{z_{N-1}}\\ r_{w_{N-1}}\\ r_{z_N}\\ \hline \vdots\\ r_{\lambda_{N-1}}\\ r_{\lambda_N}
\end{array}\right] \qquad (9)$$

where $r_{z_l} = \nabla_{z_l}\mathcal{L}$,$r_{w_l} = \nabla_{w_l}\mathcal{L}$, $r_{\lambda_l} = \nabla_{\lambda_l}\mathcal{L}$, $P_l = \nabla_{z_l z_l}\mathcal{L}$, $W_l = \nabla_{w_l w_l}$, $\mathcal{L}$, and $F_l = \nabla_{z_l w_l}\mathcal{L}$, evaluated about the current trajectory. In condensed form, the KKT system becomes,

$$\left[\begin{array}{c|c} \mathbf{H} & \mathbf{J}^T \\ \hline \mathbf{J} & \end{array}\right]
\left[\begin{array}{c} \Delta x \\ \hline \Delta\lambda \end{array}\right]
= -
\left[\begin{array}{c} r_x \\ \hline r_\lambda \end{array}\right]. \qquad (10)$$

where $\mathbf{H}$ is the Hessian matrix and $\mathbf{J}$ is the Jacobian matrix.

The formation and solution of the KKT system is the most expensive step in NLP algorithms. There exist different strategies to solve the KKT system (9) or, equivalently, (10). A popular strategy consists of a forward Riccati decomposition which exploits the natural forward structure of the dynamic model. This leads to the explicit recursion,

$$\begin{aligned}
\Delta z_N &= -\mathbf{\Pi}_N(r_{z_N} + \mathbf{M}_N^{-1} r_{\mathbf{M}_N}) && \text{(11a)}\\
\Delta \lambda_l &= \mathbf{M}_l^{-1}(\Delta z_l + r_{\mathbf{M}_l}) && \text{(11b)}\\
\Delta z_{l-1} &= -\mathbf{\Pi}_{l-1}(F_{l-1}W_{l-1}^{-1}G_{l-1}^T - A_{l-1}^T)\Delta\lambda_l + \mathbf{\Pi}_{l-1}(F_{l-1}W_{l-1}^{-1}r_{w_{l-1}} - r_{z_{l-1}} - \mathbf{M}_{l-1}^{-1}r_{\mathbf{M}_{l-1}}) && \text{(11c)}\\
\Delta w_{l-1} &= -W_{l-1}^{-1}F_{l-1}^T\Delta z_{l-1} + W_{l-1}^{-1}G_{l-1}^T\Delta\lambda_l - W_{l-1}^{-1}r_{w_{l-1}} && \text{(11d)}\\
l &= N, ..., 1 && \text{(11e)}
\end{aligned}$$

where,

$$\begin{aligned}
\mathbf{\Pi}_0 &= (P_0 - F_0W_0^{-1}F_0^T)^{-1}\\
\mathbf{M}_{l+1} &= (G_lW_l^{-1}F_l^T - A_l)\mathbf{\Pi}_l(F_lW_l^{-1}G_l^T - A_l^T) + G_lW_l^{-1}G_l^T\\
\mathbf{\Pi}_{l+1} &= (P_{l+1} + \mathbf{M}_{l+1}^{-1} - F_{l+1}W_{l+1}^{-1}F_{l+1}^T)^{-1}\\
\mathbf{\Pi}_N &= (P_N + \mathbf{M}_N^{-1})^{-1}\\
l &= 0, ..., N-1 && \text{(12)}
\end{aligned}$$

and,

$$\begin{aligned}
r_{\mathbf{M}_1} &= r_{\lambda_1} + G_0W_0^{-1}r_{w_0} - (G_0W_0^{-1}F_0^T - A_0)\mathbf{\Pi}_0(r_{z_0} - F_0W_0^{-1}r_{w_0})\\
r_{\mathbf{M}_{l+1}} &= r_{\lambda_{l+1}} + G_lW_l^{-1}r_{w_l} - (G_lW_l^{-1}F_l^T - A_l)\mathbf{\Pi}_l(r_{z_l} + \mathbf{M}_l^{-1}r_{\mathbf{M}_l} - F_lW_l^{-1}r_{w_l})\\
l &= 0, ..., N-1 && \text{(13)}
\end{aligned}$$

The computational complexity of this strategy scales as $O(N(n_z+n_w)^3)$. A second strategy to solve the KKT system is to perform a direct factorization of the full KKT system (10). This strategy allows the use of efficient linear algebra solvers such as MA27, MA57, Pardiso, among others. The complexity of this strategy scales as $O((N(n_z+n_w))^\beta),\ \beta = [1,3]$. For a fully sparse KKT matrix (as in simultaneous collocation-based approaches) the complexity of a direct factorization scales linearly and, at most, quadratically with the number of states and degrees of freedom ($\beta = 1, 2$). When dense blocks appear in the KKT matrix (as in multiple shooting) then $\beta = 3$ and a forward decomposition strategy becomes more efficient.

Notice that, depending on the approach used to solve the MHE problem, the cost of computing the

Newton step will be distributed differently between the formation of the KKT matrix and the solution of the KKT system. For single and multiple shooting the complexity is oriented towards evaluating derivative information, while with simultaneous collocation it is oriented towards the factorization of the KKT matrix. For a detailed discussion, please refer to [9].

When *exact second order information* is used to form the KKT system, the computed search step might not be a descent direction due to the presence of nonconvexities. In this case, it is necessary to regularize the Hessian matrix in order to ensure a proper direction and thus avoid convergence to saddle points or maximums. If no regularization is required at the solution of the problem, then the sufficient second order condition (SSOC) holds, and the solution is a strong minimum [15]. In the context of parameter and state estimation, this implies that the degrees of freedom can be determined *uniquely* based on the provided data [16]. This direct check for SSOC, through inertial properties of the KKT matrix, is an important advantage of using a direct factorization for the solution of large-scale and highly nonlinear problems.

3.2 NLP Sensitivity

As stated before, the MHE problem is parametric in the data contained in the parameter p. Consequently, the KKT or optimality conditions (7) are implicit functions of p and can be represented as,

$$\varphi(s(p, N), p) = 0 \tag{14}$$

where the solution vector is defined as $s(p,N)^T = [z_0^T, w_0^T, \lambda_1^T, z_1^T, w_1^T, ..., \lambda_N^T, z_N^T]$ and has optimal values $s^*(p, N)$.

In the following, we want to study the effect of perturbations $|p - p_0|$ on a neighborhood around a given nominal solution $s^*(p_0, N)$. To derive the desired results we assume that:

Assumption 1 *(NLP Sensitivity Assumptions)[17, 18]*

- *$f(\cdot)$, $L(\cdot)$ and $\Gamma(\cdot)$ are twice continuously differentiable in all their arguments. Also, these functions and their derivatives are bounded.*
- *A nominal solution $s^*(p_0, N)$ of $\mathcal{M}_N(p)$ satisfies the linear independence constraint qualifications (LICQ), sufficient second order conditions (SSOC) and strict complementary slackness.*

SSOC requires that the Hessian of the Lagrange function (6) evaluated at $s^*(p_0, N)$ is positive definite in any feasible direction [19]. A direct consequence of these assumptions is that the solution vector $s(p, N)$ becomes a continuously differentiable function in a neighborhood of a nominal solution $s^*(p_0, N)$ and $\frac{\partial s}{\partial p}\Big|_{p_0}$ is bounded and *unique* [17, 18]. This allows the application of the implicit function theorem to

(14) at $s^*(p_0, N)$ to yield:

$$\mathbf{K}^*(p_0, N) \left.\frac{\partial s}{\partial p}\right|_{p_0} = - \left.\frac{\partial \varphi(s(p, N), p)}{\partial p}\right|_{s^*(p_0,N),p_0} \tag{15}$$

where $\mathbf{K}^*(p_0, N)$ denotes the KKT matrix of $\mathcal{M}_N(p_0)$ evaluated at $s^*(p_0, N)$ with structure,

$$\mathbf{K}^*(p_0, N) = \begin{bmatrix}
P_0(p_0) & F_0(p_0) & -A_0^T & & & & & & & \\
F_0^T(p_0) & W_0 & -G_0^T & & & & & & & \\
-A_0 & -G_0 & & \mathbb{I}_{n_z} & & & & & & \\
& & \mathbb{I}_{n_z} & \ddots & & & & & & \\
& & & \ddots & \mathbb{I}_{n_z} & & & & & \\
& & & \mathbb{I}_{n_z} & P_{N-1}(p_0) & F_{N-1}(p_0) & -A_{N-1}^T & & & \\
& & & & F_{N-1}^T(p_0) & W_{N-1} & -G_{N-1}^T & & & \\
& & & & -A_{N-1} & -G_{N-1} & & & \mathbb{I}_{n_z} & \\
& & & & & & & \mathbb{I}_{n_z} & P_N(p_0) &
\end{bmatrix}$$

The most important and practical consequence of these results is that, if the nominal solution satisfies Assumption 1, then the KKT matrix is non-singular [19] and can be used to compute the sensitivity matrix from (15). In addition, these post-optimal calculations can be performed *without having to form and refactorize* the KKT matrix, which is the most dominant computational step in the NLP algorithm.

The solution of neighboring problems can be written as:

$$s^*(p, N) = s^*(p_0, N) + \left.\frac{\partial s}{\partial p}\right|_{p_0} (p - p_0) + O(|p - p_0|^2) \tag{16}$$

and fast first-order estimates of these solutions can be obtained from the explicit form,

$$\tilde{s}(p, N) = s^*(p_0, N) + \left.\frac{\partial s}{\partial p}\right|_{p_0} (p - p_0) \tag{17}$$

where $\tilde{s}(p, N)$ is an approximate solution of $s^*(p, N)$. From Assumption 1 and (16), $\tilde{s}(p, N)$ satisfies,

$$|\tilde{s}(p, N) - s^*(p, N)| \leq L_s |p - p_0|^2 \tag{18}$$

with positive constant L_s and where $|\cdot|$ is the Euclidean norm [17, 20].

Remark 1. The step $\Delta s(p, N) = \tilde{s}(p, N) - s^*(p_0, N)$ in (17) can also be found by linearization of the KKT conditions (7) around $s^*(p_0, N)$ to give,

$$\mathbf{K}^*(p_0, N) \Delta s(p, N) = -\varphi(s^*(p_0, N), p). \tag{19}$$

where the right-hand side corresponds to the KKT conditions (7) evaluated at the solution of the nominal problem. Here, $\Delta s(p,N) = \tilde{s}(p,N) - s^*(p_0,N)$ is a *perturbed* Newton step taken from $s^*(p_0,N)$ towards the solution of a neighboring problem $\mathcal{M}_N(p)$ so that $\tilde{s}(p,N)$ satisfies (17)-(18). Furthermore, computing this step requires a single fast backsolve since the KKT matrix is already factorized at the optimal solution. The computational complexity of a backsolve is $O(N(n_z+n_w))$ which is *orders of magnitude* smaller than the complexity of forming and solving the KKT system.

Remark 2. The satisfaction of SSOC is a fundamental prerequisite for the computation of NLP sensitivity information. Checking for SSOC is not usually performed directly by most NLP algorithms but can be inferred from the inertia of the KKT matrix [15, 16].

Remark 3. In the presence of bound constraints, continuity of $s(p,N)$ implies that (18) is only valid on a neighborhood of $s^*(p_0,N)$ where the current active set (set of variables at the boundary of $\mathbb{Z}$ or $\mathbb{W}$) is preserved. Assumption 1 ensures that a nonzero neighborhood exists with these properties [17]. If a change in the active set occurs due to a given perturbation $|p-p_0|$, the structure of the KKT matrix will change due to addition/dropping of active constraints. In this case, the factorization of the KKT matrix can be re-used through Schur complement techniques to correct the active set and obtain a fast approximate solution of the neighboring problem. This is equivalent to solve a quadratic programming problem efficiently.

Remark 4. If we assume $f(\cdot)$ is linear and $\Gamma(\cdot)$ and $L(\cdot)$ are quadratic functions (linear MHE) then $\tilde{s}(p,N) = s^*(p,N)$.

3.3 Extraction of Exact Covariance Information

Extracting covariance information from the solution of the MHE problem is important since it can be used as a measure of uncertainty in robust NMPC formulations [21], and to update the arrival cost [3]. In particular, note that setting $\bar{\Pi}_0(k+1) = \mathbf{M}_1$ with $\mathbf{M}_1$ extracted from (12) provides a stronger update than the traditional EKF update in (5) since it incorporates exact second order information. To see this, we extract $\mathbf{M}_1$ from (12),

$$\begin{aligned} \mathbf{\Pi}_0 &= (P_0 - F_0 W_0^{-1} F_0^T)^{-1} \\ \mathbf{M}_1 &= (G_0 W_0^{-1} F_0^T - A_0)\mathbf{\Pi}_0 (F_0 W_0^{-1} G_0^T - A_0^T) + G_0 W_0^{-1} G_0^T. \end{aligned} \tag{20}$$

If we neglect all the second order contributions of the model and the cross interaction term F_0 then from (1) and (2), $P_0 = \bar{\Pi}_0(k)^{-1} + C_0^T R_0^{-1} C_0$ and $W_0^{-1} = Q_0$. If we apply the matrix inversion lemma to P_0^{-1} we obtain,

$$\begin{aligned} \mathbf{\Pi}_0 &= \bar{\Pi}_0(k) - \bar{\Pi}_0(k) C_0^T (R_0 + C_0 \bar{\Pi}_0(k) C_0^T)^{-1} C_0 \bar{\Pi}_0(k) \\ \mathbf{M}_1 &= A_0 \mathbf{\Pi}_0 A_0^T + G_0 Q_0 G_0^T \end{aligned} \tag{21}$$

which is the simplified EKF update formulate (5). In classical estimation literature, the simplified matrices $\mathbf{M}_l,\ l = 1, ..., N$ are known as the *prior* covariance matrices [22]. Using similar simplifications, we can show that matrices $\mathbf{\Pi}_l,\ l = 1, ..., N$ in (12) become,

$$\mathbf{\Pi}_l^{-1} \quad = \quad \mathbf{M}_l^{-1} + C_l^T R_l^{-1} C_l, \qquad l = 1, ..., N. \tag{22}$$

These are normally known as the *posterior* covariance matrices where $\mathbf{\Pi}_N$ is the covariance matrix of the current state estimate. The exact representation of this matrix can be obtained from recursion (12).

If a Riccati decomposition is used to solve the KKT system in the NLP algorithm, the exact covariance matrices $\mathbf{\Pi}_N$ and $\mathbf{M}_1$ are obtained as a natural outcome. However, if a direct factorization is used, these matrices are never formed. Instead, note from (11a) that if we set $r_{z_N} = -\mathbb{I}_{n_z}(:, j)$ in (9) at the solution (all right hand sides vanish and $r_{\mathbf{M}_N} = 0$) where $\mathbb{I}_{n_z}(:, j)$ is the j-th column of the identity matrix then, $\Delta z_N = \mathbf{\Pi}_N(:, j)$. This implies that the exact covariance matrix can be computed efficiently by performing n_z backsolves with the already factored KKT matrix from (9) or, equivalently, (10). Extracting $\mathbf{M}_1$ from the KKT matrix is not as straightforward. However, from (9) and (11e) it is possible to prove that, setting all the right-hand sides in (9) to zero and forcing $\Delta z_1 = -\mathbb{I}_{n_z}(:, j)$ recursively, leads to $\Delta\lambda_1 = \mathbf{M}_1^{-1}(:, j)$. In other words, it is possible to extract the covariance matrices at different time steps from the KKT matrix through backsolves with the factorized KKT matrix.

4 Fast MHE Algorithm

The parametric properties of the MHE problem can be exploited to design fast MHE strategies able to remove the on-line computational delay. Assume that at time t_k we have $x(k)$, $\bar{z}_0(k)$, $\bar{\Pi}_0(k)^{-1}$, and the measurements $\{y(k-N)...y(k)\}$. We would like to use this information to obtain a fast approximation of the state estimate $x(k+1)$ at t_{k+1} but we do not yet have the future measurement $y(k+1)$. In order to overcome this, we propose the following sensitivity-based shifting strategy:

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

1. Compute a disturbance-free ($w = 0$) extrapolation of the current state $\bar{x}(k+1) = f(x(k), 0)$ and corresponding output $\bar{y}(k+1) = h(\bar{x}(k+1))$ through forward simulation.

2. Define the extended data $\bar{\eta}(k+1) = (\bar{z}_0(k), \bar{\Pi}_0^{-1}(k), y(k-N)...y(k), \bar{y}(k+1))$ and solve the *extended* problem $\mathcal{M}_{N+1}(p_0)$ with $p_0 = \bar{\eta}(k+1)$ and $N+1$ time steps.

3. At the solution $s^*(p_0, N+1)$, hold KKT matrix $\mathbf{K}^*(p_0, N+1)$ or compute sensitivity matrix $\left.\frac{\partial s}{\partial p}\right|_{p_0}$.

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

1. Obtain the true measurement $y(k+1)$ and define the *true* problem data $\eta(k+1)$. Compute an instantaneous approximate solution $\tilde{s}(p, N+1)$ from sensitivity (17) or as a perturbed Newton step (19) and extract $\tilde{x}(k+1) = \tilde{z}_{N+1}$ and $\bar{z}_0(k+1) = \tilde{z}_1$. Extract $\bar{\Pi}_0(k+1) = \mathbf{M}_1$ from the fixed KKT matrix $\mathbf{K}^*(p_0, N+1)$.

2. Update data, set $k := k+1$ and return to the background process.

If the above strategy is used, the NLP sensitivity perturbation is $|p - p_0| = |y(k+1) - \bar{y}(k+1)|$ so that the approximation error is $|\tilde{x}(k+1) - x(k+1)| = O|y(k+1) - \bar{y}(k+1)|^2$. In addition, since the perturbation is small, the approximation error of the perturbed Newton step is expected to be of the same order.

Note that the covariance matrix $\bar{\Pi}_0(k+1)$ is extracted from $\mathbf{K}^*(p_0, N+1)$ which is evaluated at $s^*(p_0, N+1)$. The optimal covariance matrix should be extracted from $\mathbf{K}^*(p, N+1)$ evaluated at $s^*(p, N+1)$. From Lipschitz continuity of the KKT matrix (Assumption 1) it can be shown that $|\mathbf{K}^*(p, N+1) - \mathbf{K}^*(p_0, N+1)| \leq L_s|p - p_0|$ for a given positive constant L_s. That is, the error bound between the approximated and optimal covariance matrices can be shown to be $O|y(k+1) - \bar{y}(k+1)|$.

From the optimality conditions (7) of the extended problem $\mathcal{M}_{N+1}(\bar{\eta}(k+1))$ we can show that $\lambda^*_{N+1} = 0$ since, by construction, $\bar{y}(k+1) = h_{N+1}(z_{N+1})$. In addition, this implies that $w^*_N = 0$. If we compute the perturbed Newton step (19) through a Riccati recursion, the last element of the recursion becomes,

$$\begin{aligned} \Delta z_{N+1} &= -\mathbf{\Pi}_{N+1} r_{z_{N+1}} \\ &= \mathbf{\Pi}_{N+1} C^T_{N+1} R^{-1}_{N+1} \left(\bar{y}(k+1) - h_{N+1}(z^*_{N+1}(\bar{y}(k+1))) \right). \end{aligned} \tag{23}$$

Once the new measurement $y(k+1)$ is obtained, we perturb the right hand side so that,

$$\begin{aligned} \Delta z_{N+1} &= \tilde{z}_{N+1}(y(k+1)) - z^*_{N+1}(\bar{y}(k+1)) \\ &= \mathbf{\Pi}_{N+1} C^T_{N+1} R^{-1}_{N+1} (y(k+1) - h_{N+1}(z^*_{N+1}(\bar{y}(k+1))) \\ &= K_{N+1}(y(k+1) - h_{N+1}(z^*_{N+1}(\bar{y}(k+1))). \end{aligned} \tag{24}$$

This expression can be seen as an analog of the Kalman Filter update formula where K_{N+1} is the so-called Kalman matrix [22]. Here, $z^*_{N+1}(\bar{y}(k+1))$ can be interpreted as the before-measurement state estimate while $\tilde{x}(k+1) = \tilde{z}_{N+1}(y(k+1))$ can be interpreted as the after-measurement estimate. The above expression is also useful to analyze the impact of the characteristics of the dynamic system and of the design of the MHE problem on the approximation error. For instance, it is clear that as $N \to \infty$ (i.e.; we add more information) then $\mathbf{\Pi}_{N+1} \to 0$. Therefore, the impact of the update becomes negligible and the approximation error $|\tilde{x}(k+1) - x(k+1)| \to 0$. Similarly, if the system is strongly

observable then the eigenvalues of $\mathbf{\Pi}_{N+1}$ will be positive and small and the approximation error will also tend to be small.

Remark 1. The proposed fast MHE algorithms assume that the background problem $\mathcal{M}_{N+1}$ can be solved within one sampling time. Fast MHE strategies able to accommodate larger background MHE problems, extending over multiple sampling times by reusing the KKT matrix in multiple time steps, have been presented in [13].

Remark 2. The proposed fast MHE strategy is applicable to different MHE formulations such as robust MHE. The only requirement of a given formulation is continuous differentiability of the optimality conditions in order to guarantee the existence of bounded approximation errors.

5 Case Studies

We illustrate the developments using two simulated case studies. In the first small-scale case we emphasize on the performance of the fast MHE algorithm while in the second large-scale case we emphasize on the computational performance of the algorithm. In both cases, the continuous-time MHE problems are discretized using orthogonal collocation on finite elements. The elements are placed in order to match the sampling times. The resulting NLPs are solved with the full-space NLP solver IPOPT [23]. The solver has been equipped with covariance information extraction and NLP sensitivity capabilities. This allows the implementation of fast MHE strategies for large-scale applications.

5.1 Nonlinear Continuous Stirred Tank Reactor

We consider a simulated MHE scenario on the nonlinear continuous stirred tank reactor (CSTR) presented by Hicks and Ray [24]:

$$\frac{dz^c}{dt} = \frac{z^c(t)-1}{\theta} + k_0 \cdot z^c(t) \cdot \exp\left[\frac{-E_a}{z^t(t)}\right] \tag{25a}$$

$$\frac{dz^t}{dt} = \frac{z^t(t)-z^t_f}{\theta} - k_0 \cdot z^c(t) \cdot \exp\left[\frac{-E_a}{z^t(t)}\right] + \alpha \cdot u(t) \cdot (z^t(t)-z^t_{cw}) \tag{25b}$$

The system involves two states $z = [z^c,\, z^t]$ corresponding to the concentration and temperature, and one control u corresponding to the cooling water flowrate. The model parameters are $z^t_{cw} = 0.38$, $z^t_f = 0.395$, $E_a = 5$, $\alpha = 1.95 \times 10^4$, and $k_0 = 300$.

The objective of this case study is to analyze the effect of NLP sensitivity errors on the performance of the fast MHE estimator and contrast this with the performance of the optimal or ideal MHE estimator. We use batch data generated from a simulated NMPC scenario.

As a first result, we compare the effect of the estimation horizon length N on the covariance matrix $\mathbf{\Pi}_N$ of the state estimates when the temperature z^t is used as measurement to infer the concentration z^c. The resulting 95% confidence regions of the estimate states $z^c_N\,, z^t_N$ are presented in Figure 2. The

shortest horizon used contains 5 time steps (inner ellipsoid) while the longest contains 50 time steps (outer ellipsoid). It is clear that the eigenvalues of the covariance matrix tend to zero as the horizon is increased. It has been found that the temperature z^t is the most informative measurement around the nominal operating point.

Scenario 1. The simulated states are corrupted with Gaussian noise ($\sigma = 0.01$) and we use $\bar{z}_0 = [0.3\ 0.3]$ and $\bar{\Pi}_0^{-1} = \text{diag}\{0.1,0.1\}$ as initial guess. The weights for the output deviations are set to $\frac{1}{\sigma^2}$. In Figure 3 we compare the performance of both the ideal and fast MHE algorithms. As can be seen from from subplot b), both algorithms are able to reconstruct the true state z^c of the system based on temperature information. The fast MHE estimator is able to remove computational delay. Performance deterioration of fast MHE due to NLP sensitivity errors is not immediately evident from these profiles but can be appreciated by comparing the cost functions of both MHE algorithms in subplot c). It is interesting to observe that the performance of the fast MHE algorithm degrades at some time steps creating small deviations from the optimal cost function of the ideal MHE. This is due to the fact that, at these points, the difference between the predicted and actual measurement is quite large, leading to large approximation errors. The difference in performance tends to disappear as information is accumulated and identical performance is observed for the rest of the operating horizon even in the presence of large levels of noise. These results are in agreement with equation (24) and the observations of Section 4.

Scenario 2. We add two disturbances on the activation energy E_a to test the robust performance of both MHE estimators. As can be seen from subplots a) and b) in Figure 4, the disturbance jumps at time step 75 and 100 disrupt the inferred state profile, but the estimators are able to reject the disturbances. Interestingly, from the cost functions of subplot c) we see that the effect of approximation errors does not degrade the performance of the fast MHE estimator significantly.

5.2 Full-Scale LDPE Plant

We consider a simulated MHE scenario arising on a full-scale low-density polyethylene (LDPE) process. A simplified flowsheet of a typical LDPE plant is depicted in Figure 5. For a more detailed explanation of the process, please refer to [16] and the references therein. In this process, ethylene is polymerized in a long tubular reactor at high pressures (2000-3000 atm) and temperatures (150-300 ^{o}C) through a free-radical mechanism. The final product is recovered by flash separation. The process presents a difficult dynamic system with long time delays due to the recycle loops and multiple compression stages. The only available measurements are the reactor temperature profile and the gas concentration leaving the hyper-compressor. The objective is to estimate the remaining differential states corresponding to the concentrations of ethylene, butane, methane and impurities throughout the plant units. The dynamic

evolution of these states can be described by material balances around each plant unit,

$$\begin{aligned} \frac{d\left(V_k \cdot \rho_k \cdot z_{k,j}\right)}{dt} &= F_k\, z_{k,j}^{in} - F_k z_{k,j} \\ z_{k,j}(0) &= z_0^{k,j} \\ k = 1, ..., N_U, \quad j &= 1, ..., N_C \end{aligned} \tag{26}$$

where N_C, number of gaseous components in the process; N_U, number of plant units; F_k, mass flow rate (kg/h); V_k, equipment volume (m^3); t, time (s); ρ_k, gas density (kg/m^3); $z_{k,j}^{in}$, inlet weight component of j-th component to the k-th unit; $z_{k,j}$, outlet weight component of j-th component from the k-th unit. The gas density at the extreme conditions is calculated through nonlinear thermodynamic relations. Most of the complexity of the dynamic model is caused by the presence of time delays which are modeled by material balances around pipes of fixed length:

$$\frac{\partial z_{i,j}}{\partial t} + \frac{1}{\tau_i}\frac{\partial z_{i,j}}{\partial \ell} = 0, \qquad z_{i,j}(\ell, 0) \;=\; z_0^{i,j} \tag{27}$$

where τ_i represents the i-th time delay in the process with $i = 1, ..., N_T$ and $j = 1, ..., N_C$. The PDEs are transformed to ordinary differential equations by applying a spatial finite difference scheme with 10 intervals. The resulting large-scale DAE model contains 294 differential and 64 algebraic state variables. The concentration of butane in the recycle loop y_{C_4} is used as the only measured output. The output measurements were obtained by simulation of the dynamic model using fixed control profiles over a long horizon of 5.6 hours divided into 60 sampling points. The predicted output profile is then corrupted using Gaussian noise with $\sigma = 0.05$. Following this reasoning, the least-squares objective function,

$$\min_{\mathbf{z}_0} \quad (\mathbf{z}_0 - \bar{\mathbf{z}}_0)^T \bar{\Pi}_0^{-1} (\mathbf{z}_0 - \bar{\mathbf{z}}_0) + \sum_{l=0}^{N} \frac{1}{\sigma^2} (y_{C_4}(t_k) - y_{C_4}(k - N + l))^2 \tag{28}$$

and the model equations (26)-(27) are used for the formulation of the estimation problem. Here, $y_{C_4}(k)$ is the measured concentration of butane in the recycle loop at sampling time t_k, vector $\mathbf{z}_0 \in \Re^{294}$ contains the initial conditions for all the states with a given *a priori* estimate $\bar{\mathbf{z}}_0$ obtained from simulation. Finally, $\bar{\Pi}_0^{-1} \in \Re^{294 \times 294}$ is a diagonal matrix with entries set to $\frac{1}{0.1}$ and we impose lower and upper bounds on all the states. We use a total of 15 finite elements and 3 collocation points for time discretization of the dynamic model. The resulting NLP contains 27,121 constraints, 9330 lower bounds, 9330 upper bounds and 295 degrees of freedom corresponding to the initial conditions for the states and an extra dummy variable. Since the dynamics of the system are slow, a total horizon time of 1.4 hours is used with sampling times $(t_{\ell+1} - t_\ell) = 5.6$ min.

Computational results associated to the solution of the NLP using IPOPT are presented in Table 1. It is clear that the vast majority of the total CPU time is devoted for the factorization of the KKT

matrix. A standard MHE algorithm solving the MHE problem on-line would introduce an estimation delay of more than 3 minutes, while the fast MHE algorithm only needs to perform a single backsolve on-line, which takes less than 1 second.

It has been found that SSOC holds at the solution of the MHE problems. Therefore, it is possible to conclude that the state of the system is observable given the limited measurement data. Figure 6 presents the measured, estimated and true profiles of the output variable along 60 sampling times. The fast MHE algorithm is able to estimate accurately the true output variable. The noise perturbations do not induce drastic changes between neighboring problems. As a consequence, a fast backsolve is required to obtain nearly instantaneous and accurate state estimates.

The approximate solutions obtained from NLP sensitivity are used to warm-start the algorithm for the solution of the background nominal problems at each sampling time. By doing so, the algorithm is able to converge the background problems using only 3-5 iterations.

6 Conclusions

A fast moving horizon estimation algorithm is presented in this work. The recursive strategy solves a nominal problem in between sampling times using a predicted future measurement and corrects on-line using fast nonlinear programming sensitivity calculations. Rigorous performance bounds are derived based on classical NLP sensitivity results. A detailed analysis of the optimality conditions of the MHE problem is performed in order to derive strategies for fast covariance information extraction for general NLP algorithms. It is demonstrated through simulation studies that the proposed algorithm is able to mimic the performance of ideal MHE. As part of future work, we propose to incorporate the derived approximation bounds in a detailed stability analysis of fast MHE algorithms. Finally, we propose to study the interaction between fast MHE algorithms and recently proposed fast sensitivity-based NMPC algorithms [9, 10].

References

[1] Michalska, H. and D. Q. Mayne (1995). Moving horizon observers and observer-based control. *IEEE Trans. Automat. Contr.* **40**, 995-1006.

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

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

[4] Rawlings, J. B. and B. R. Bakshi (2006). Particle filtering and moving horizon estimation. *Comp. Chem. Eng.* **30**, 1529-1541.

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

[6] Findeisen, R. and F. Allgöwer (2004). Computational delay in nonlinear model predictive control. In: *Proc. Int. Symp. Adv. Control of Chemical Processes, ADCHEM 03*. Hong Kong.

[7] Chen, W., D.J. Balance, and J. O'Reilly (2000). Model predictive control of nonlinear systems: Computational burden and stability. *IEEE Proc. Control Theory Appl.*, **147**, 387–394.

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

[9] Zavala, V.M., C. D. Laird and L. T. Biegler (2007). Fast implementations and rigorous models: Can both be accomodated in NMPC? *Int. J. Robust Nonlinear Control*, In Press.

[10] Zavala, V.M., and L. T. Biegler (2007). The Advanced Step NMPC Controller. Stability, Optimality and Robustness. *Submitted for Publication.*

[11] Kraus, T., P. Kuehl, L. Wirsching, H. S. Bock and M. Diehl (2006). A moving horizon state estimation algorithm applied to the Tennessee Eastman benchmark process. In: *IEEE Conference on Multisensor Fusion and Integration*. Heidelberg, Germany.

[12] Diehl, M., R. Findeisen, H.G. Bock, J.P. Schlöder and F. Allgöwer (2005*b*). Nominal stability of the real-time iteration scheme for nonlinear model predictive control. *IEEE Control Theory Appl.* **152, 3**, 296–308.

[13] Zavala, V.M., C.D. Laird and L.T. Biegler (2007). A Fast Computational Framework for Large-Scale Moving Horizon Estimation. In *Proceedings of 8th International Symposium on Dynamics and Control of Process Systems*, Cancun, Mexico.

[14] Poku, M.Y.B. and L.T. Biegler. (2004) Nonlinear Optimization with Many Degrees of Freedom in Process Engineering. *Ind. Eng. Chem. Res.*, **43**, 6803-6810.

[15] Forsgren, A., P.E. Gill and M.H. Wright (2002). Interior methods for nonlinear optimization. *SIAM Review* **44**, 525–597.

[16] Zavala, V.M. and L.T. Biegler (2006). Large-scale parameter estimation in low-density polyethylene tubular reactors. *Ind. Eng. Chem. Res.* **45**, 7867–7881.

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

[18] Fiacco, A. V. (1976) Sensitivity Analysis for Nonlinear Programming Using Penalty Methods. *Math. Programm.*, **10**, 287–311.

[19] Nocedal, J. and S. J. Wright. (1999) *Numerical Optimization*. New York, Springer-Verlag.

[20] Dennis, J. E. and R. B. Schnabel. (1996) *Numerical Methods for Unconstrained Optimization and Nonlinear Equations*. Philadelphia, SIAM.

[21] Nagy, Z.K. and R.D. Braatz. (2003) Worst-Case and Distributional Robustness Analysis of Finite-Time Control Trajectories for Nonlinear Distributed Parameter Systems. *IEEE Tras. Cont. Sys. Theory*, **11**, 694–704.

[22] A. E. Bryson and Y. C. Ho. (1975) *Applied Optimal Control*. Taylor & Francis.

[23] Wächter, A. and L.T. Biegler (2006). On the implementation of a primal-dual interior point filter line search algorithm for large-scale nonlinear programming. *Math. Program.* **106**, 25–57.

[24] Hicks, G.A. and W.H.Ray (1971) Approximation methods for optimal control synthesis. *Can. J. Chem. Eng.*, **49**, 522–529.

Algorithmic Step	**CPU Time (s)**
Full Solution (6 iterations)	202.64
Single Factorization of KKT Matrix	33.77
Step Computation (single backsolve)	0.9-1.0
Rest of Steps	0.936

Table 1: Average computational times associated to the background MHE solution for the LDPE case study (3.0 GHz Pentium IV processor, 1 Gb RAM).

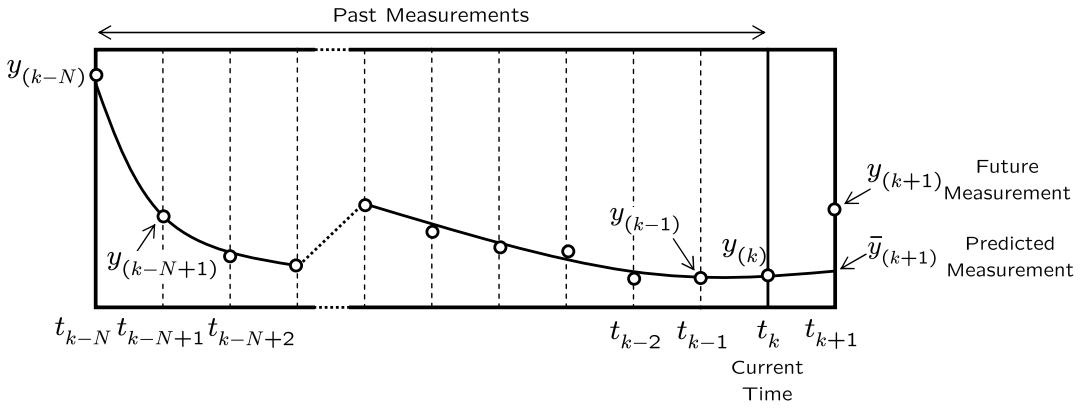

Figure 1: Schematic representation of estimation horizon for extended problem $\bar{\mathcal{M}}_{N+1}(\eta(k))$.

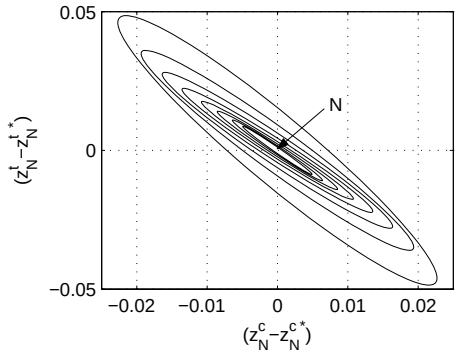

Figure 2: Effect of estimation horizon length N on the state estimates covariance when z^t is used as measurement.

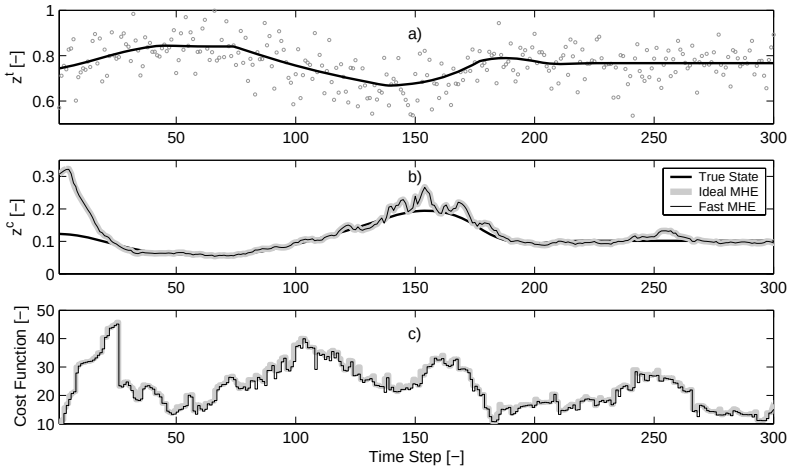

Figure 3: Scenario 1. Comparison of performance of fast and ideal MHE strategies in the presence of large measurement noise.

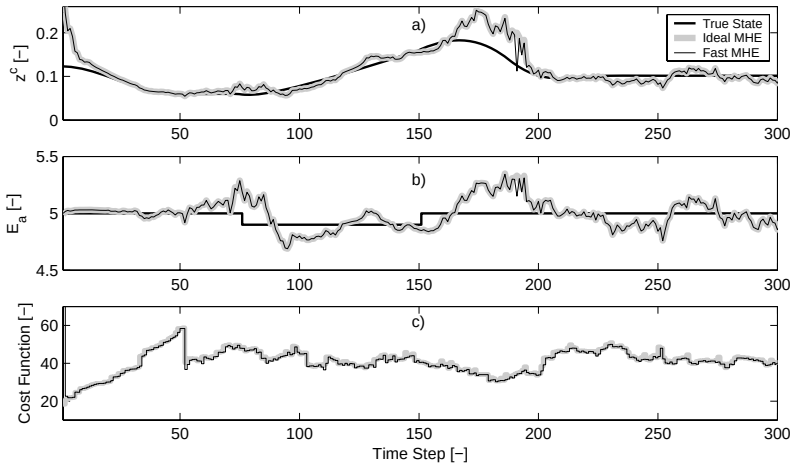

Figure 4: Scenario 2. Comparison of performance of fast and ideal MHE strategies in the presence of noise and disturbances.

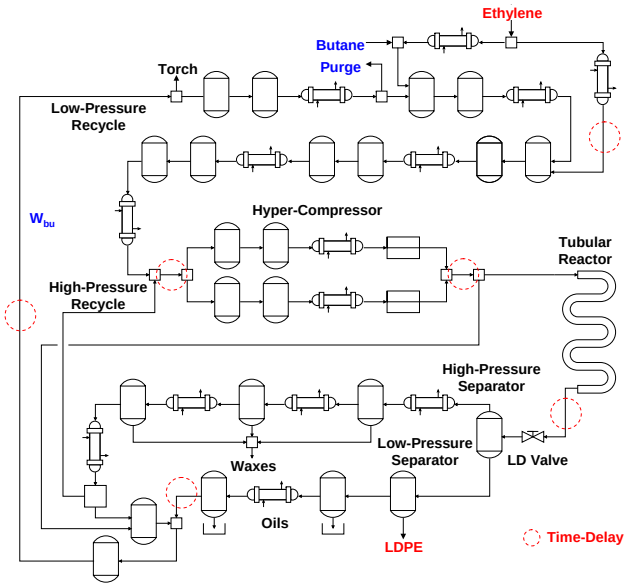

Figure 5: Simplified flowsheet of a typical high-pressure LDPE tubular reactor process.

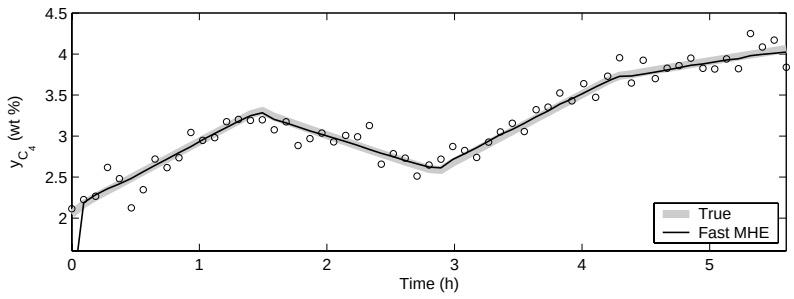

Figure 6: Measured, estimated and true profiles of the output variable for LDPE case study.