MODEL-PREDICTIVE CONTROL AND CLOSED-LOOP STABILITY CONSIDERATIONS FOR NONLINEAR PLANTS DESCRIBED BY LOCAL ARX-TYPE MODELS

Michael E. Cholette
School of Chemistry, Physics, Mechanical Engineering
Queensland University of Technology
Brisbane, Queensland, Australia
Email: michael.cholette@qut.edu.au

Dragan Djurdjanovic
Department of Mechanical Engineering
University of Texas at Austin
Austin, Texas
Email: dragand@me.utexas.edu

ABSTRACT

In this paper, a model-predictive control (MPC) method is detailed for the control of nonlinear systems with stability considerations. It will be assumed that the plant is described by a local input/output ARX-type model, with the control potentially included in the premise variables, which enables the control of systems that are nonlinear in both the state and control input. Additionally, for the case of set point regulation, a suboptimal controller is derived which has the dual purpose of ensuring stability and enabling finite-iteration termination of the iterative procedure used to solve the nonlinear optimization problem that is used to determine the control signal.

1 INTRODUCTION

Control design based on Local Dynamic Models (LDMs) has attracted significant interest in the past two decades. The copious literature on controller design for Takagi-Sugeno Fuzzy Models (TSFMs) and Piecewise Affine (PWA) models is a testament to the the appeal of LDMs for their generality, interpretability and potential for relating well-known linear techniques to nonlinear systems.

One popular controller design technique with stability conditions is Parallel Distributed Compensation (PDC) [1–3]. PDC utilizes TSFMs with a local linear structure and each region possesses a local linear controller. A set of Linear Matrix Inequalities (LMIs) is formulated to solve for a set of stabilizing local gains and demonstrate the stability of the closed-loop system. However, the PDC approach typically requires that the local models be absent of bias terms, which has been found to lead to significantly diminished approximation capability [4]. Control of the more general (PWA) systems have been the subject of a number of papers [5–8], which also utilize LMIs. However, for both PWA and PDC, the nonlinear controller design is restricted to premise variables which are typically independent of the control, and are thus limited to systems that are linear in the control input [9].

To relax such restrictions, model-predictive control (MPC) strategies have proven useful. Recently, a number of strategies using neuro-fuzzy approaches have been proposed that require milder assumptions on the underlying model. Roubos et al. [10] utilized a TSFM and a model-predictive strategy based on local linearization of the TSFM, which led to a quadratic program (QP) subproblem. However, no stability analysis was conducted and for a prediction horizon, N, the multi-step linearization strategy required the solution of N QPs for each sample time. Other TSFM strategies were employed in Abonyi et al. [11], where the authors use an instantaneous linearization, and Li et al. [12] who use a linear generalized predictive controller (GPC) in each TSFM region, after which they heuristically blend the outputs of each regional controller. In a similar approach, Hadjili and Wertz [13] proposed a TSFM-based GPC controller and considered a number of approaches for deriving a solution to a MPC problem. Fischer et al. [14] use a local model structure to describe a GPC and Dynamic Matrix Control (DMC) approach. Unfortunately, none of these approaches addressed stability of the resulting closed-loop system.

Dziekan et al. [15] controlled a discrete-time state-space TSFM using a MPC strategy augmented with a PDC law. The LMIs solved for the PDC feedback law were used for a warm start for the subsequent time step optimization. Zhang et al. [16] used a TSFM to perform robust output feedback control. Their methodology utilizes a number of LMIs for the observer and controller design, which are obtained from the Piecewise Quadratic Lyapunov functions (PWQLFs) used to ensure stability. However, for TSFMs with a large number of regions, the LMI conditions can become computationally expensive and, due to the conservativeness of PWQLFs, potentially infeasible, even for stable systems.

A number of other neural approaches have been employed recently. Sørensen et al. [17] implemented a generalized predictive controller using a multi-layer perceptron network as a Nonlinear AutoRegressive eXogenous (NARX) model. The authors then employed a Newton-type optimization routine to minimize the cost function over the prediction horizon, but did not discuss under what conditions their algorithm is stable. Lu and Tsai [18] utilize a Nonlinear AutoRegressive eXogeneous (NARX) model structure composed of local AutoRegressive eXogenous (ARX) models to derive a predictive control law for single-input single-output systems. The authors consider the stability of the closed-loop system, but the control horizon is limited to one. Nikravesh et al. [19] employed a neural network version of DMC which utilizes a neural approximation of the underlying system, which is similar to the step response model of DMC. The authors then utilize the one step ahead prediction in a MPC scheme and discuss stability for open-loop stable plants.

In this paper, a MPC controller design procedure will be developed for the control of a plant described by a local ARX model structure [20, 21], with stability considerations for an arbitrary MPC horizon. Formulation of the MPC procedure will be such that the feasibility of the control signal solution will yield a stable closed-loop system. Furthermore, determination of this feasibility can be conducted in the neighborhood of a proposed trajectory. Such stability analysis is often neglected in MPC approaches, particularly in the neuro-fuzzy field. In addition, the proposed approach will only use signals that are directly sensed (inputs and outputs) and the proposed approach is applicable to local models that have the control included in the premise variables, thus enabling the control of systems that are nonlinear in the control input. Finally, for the case of set point regulation, a suboptimal controller design procedure will be derived that enables finite termination of the iterative procedure used to solve the stabilizing optimal control problem, while ensuring stability of the controlled system.

The main contributions of this paper is a complete, stable MPC algorithm for a highly-general local dynamic model structure. To achieve this, a number of theoretical and practical issues are addressed. In Section 2, the generic LDM is re-expressed in a state-space form, enabling the exploitation of some MPC stability results and nonlinear optimization methods from literature, which are used to formulate the control algorithm in Section 3. In Section 4, some important practical issues arising in MPC are addressed. A suboptimal control algorithm is developed by combining the concept of *dual-mode* MPC [22] with the nonlinear optimization procedure, and using the LDM to estimate the necessary constraint regions. Finally, an example application is presented and the conclusions are discussed in Sections 5 and 6, respectively.

1.1 Notation

In the sequel, the following notational conventions will be adopted. Bold face symbols will denote sequences. For instance, the symbol $\mathbf{u} = \{u(0), u(1), \ldots, u(N-1)\}$ will denote a control sequence for time samples up to $N-1$, where N is the length of the horizon. The symbol $\mathbf{u}(k) = \{u(k|k), u(k+1|k), \ldots, u(k+N-1|k)\}$ denotes a control sequence starting at the current time, k, up to the future time $k+N-1$. Also, $\mathbf{x} = \{x(0), x(1), \ldots, x(N)\}$ will denote a state sequence of length $N+1$ and an asterisk $(*)$ will be used to denote optimal quantities. Finally, for a matrix $A \in \mathbb{R}^{n\text{x}n}$, $A > 0$ denotes definiteness.

2 PRELIMINARIES

2.1 Model Structure and Assumptions

In this paper, the local ARX model structure similar to Liu et al. [21] will be utilized for the controller design. The model structure is

$$
\begin{aligned}
y(k+1) &= \sum_{i=1}^{M} \tilde{\nu}_i(w(k)) \tilde{F}_i(s(k)) \\
\tilde{F}_i(s(k)) &= Q_i s(k) + b_i
\end{aligned}
\tag{1}
$$

where

$$
\begin{aligned}
s(k) = \big[&y^T(k), y^T(k-1), \ldots, y^T(k-n_a), u^T(k), u^T(k-1), \\
&\ldots, u^T(k-n_{bu}), d^T(k), d^T(k-1), \ldots, d^T(k-n_{bd})\big]^T
\end{aligned}
$$

where $u(k-i) \in \mathbb{R}^m$, $i = 0, 1, \ldots n_{bu}$ are the controlled system inputs, $d(k-j) \in \mathbb{R}^q$, $i = 0, 1, \ldots n_{bd}$ are the measured disturbances, $y(k-\ell) \in \mathbb{R}^p$, $\ell = 0, 1, \ldots n_a$ are the past system outputs, and $w(k)$ is composed of a subset of the elements of $s(k)$, which can include the current input, $u(k)$. The local validity functions, $\tilde{\nu}_i$, satisfy $\tilde{\nu}_i \geq 0$, and $\sum_{i=1}^{M} \tilde{\nu}_i = 1$ are assumed to be twice differentiable.

The control goal will be the tracking of a reference signal, $r(k)$. In the sequel, the following assumptions will be made.

Assumption 2.1. *(Adequacy of the LDM). The model orders n_a, n_{bu} and n_{bd} are known and Eq.*(1) *represents the true system dynamics.*

Assumption 2.2. *(Knowledge of the uncontrolled inputs). The future values of the uncontrolled inputs $d(i)$ $i \geq k$ are known.*

Let us discuss these assumptions briefly. Assumptions 2.1 and 2.2 imply that this paper will be focused on deriving a control law for the nominal case, where the model is taken as the true system and the uncontrolled inputs are known exactly for the horizon. Considerations related to uncertainty in the model and/or the uncontrolled input are outside the scope of this paper.

Note that no restriction has been made as to how the model (1) is obtained. In [21], the local dynamic model was identified from input-output data, however the controller design can proceed with local model structures identified in different ways (such as LOLIMOT models [23]) or from off-equilibrium linearizations of physical models.

2.2 Transformation to a Discrete State-Space Form

MPC methods are typically proposed using a variety of models. Examples include impulse or step response, linear/nonlinear state space, nonlinear ARX, and fuzzy models [24, 25]. While input/output models are frequently used for MPC, rigorous stability analysis is typically conducted with state-space representations [24, 26]. To this end, we convert Eq. (1) to a nonlinear state-space representation. Defining the state vector as

$$\begin{aligned} z(k) &\triangleq \left[z_y^T \,\vdots\, z_u^T \,\vdots\, z_d^T \right]^T \\ z_y &= \left[y^T(k) \;\; y^T(k-1) \;\; \dots \;\; y^T(k-n_a) \right]^T \\ z_u &= \left[u^T(k-1) \;\; u^T(k-2) \;\; \dots \;\; u^T(k-n_{bu}) \right]^T \\ z_d &= \left[d^T(k-1) \;\; d^T(k-2) \;\; \dots \;\; d^T(k-n_{bd}) \right]^T \end{aligned} \tag{2}$$

where $z(k) \in \mathbb{R}^n$ and $n = p(n_a+1) + mn_{bu} + qn_{bd}$. We can redefine Eq.(1) as

$$\begin{aligned} z(k+1) = f_z(z,u,d) &\triangleq \sum_{i=1}^{M} \hat{\nu}_i(z,u,d)\hat{F}_i(z,u,d) \\ \hat{F}_i(z,u,d) &= \hat{A}_i z + \hat{B}_i u + \hat{G}_i d + \hat{b}_i \end{aligned} \tag{3}$$

where $\hat{A}_i$, $\hat{B}_i$, $\hat{G}_i$ and $\hat{b}_i$ are defined in the appendix. Using knowledge of the uncontrolled input, system (3) can now be transformed into a nonlinear, time-varying system.

Proposition 2.1. *(Elimination of the uncontrolled input). Given the past uncontrolled inputs* $z_d(k) = \left[d^T(k), d^T(k-1), \dots, d^T(k-n_{bd})\right]^T$*, the definition of the state in Eq.* (2) *and Assumption 2.2, Eq.* (3) *can be written as*

$$x(j+1) = f(j,x,u) \quad \forall j \geq k \tag{4}$$

where the state vector is defined as $x = \left[z_y^T \quad z_u^T\right]^T$ *and*

$$f(j,x,u) = \sum_{i=1}^{M} \nu_i(j,x,u)F_i(j,x,u) \tag{5}$$

$$F_i(\ell,x,u) = A_i x + B_i u + b_i(\ell) \tag{6}$$

$$A_i = \begin{bmatrix} A_{11}^{(i)} & A_{12}^{(i)} \\ 0 & A_{22}^{(i)} \end{bmatrix} \quad B_i = \begin{bmatrix} B_1^{(i)} \\ B_2^{(i)} \end{bmatrix} \tag{7}$$

$$b_i(\ell) = \begin{bmatrix} A_{13}^{(i)} \\ \hdashline 0 \end{bmatrix} z_d(\ell) + \begin{bmatrix} G_1^{(i)} \\ \hdashline 0 \end{bmatrix} d(\ell) + \begin{bmatrix} b_1^{(i)} \\ \hdashline 0 \end{bmatrix} \tag{8}$$

$$\nu_i(\ell,x,u) = \hat{\nu}_i\left(\begin{bmatrix} x \\ \hdashline z_d(\ell) \end{bmatrix}, u, d(\ell) \right) \tag{9}$$

$$z_d(\ell+1) = A_{33}^{(i)} z_d(\ell) + G_3^{(i)} d(\ell). \tag{10}$$

where the sub-matrices A_{jk}*,* B_ℓ*,* G_m *and* b_n *are defined in the appendix.*

Remark 2.2. *If it is assumed that* $d(k) = d$, $\forall k$ *or if there are no uncontrolled inputs, then Eq.* (10) *results in a constant* $z_d(k+j) = z_d(k) = z_d$ *which implies that Eq.* (4) *is time invariant, i.e.* $x(j+1) = f(x,u)$.

Proposition 2.3. *(Linearization about a trajectory). Let the dynamics of the nonlinear system at sample time k evolve according to Eq.* (4)*, let the initial state be* $x(i)$ *and let the future input be* $\mathbf{u}^0 = \left\{u^0(i), u^0(i+1), \dots, u^0(i+N)\right\}$*. The* linearization *of this system from time* $k = i$ *to* $k = i+N$ *can be obtained as*

$$\bar{x}(k+1) = A(k)\bar{x}(k) + B(k)\bar{u}(k)$$

where

$$\begin{aligned} A(k) &\triangleq \frac{\partial f}{\partial x} = \sum_{m=1}^{M} F_m(k,x,u)\frac{\partial \nu_m}{\partial x} + \nu_m(k,x,u)A_m \\ B(k) &\triangleq \frac{\partial f}{\partial u} = \sum_{m=1}^{M} F_m(k,x,u)\frac{\partial \nu_m}{\partial u} + \nu_m(k,x,u)B_m \end{aligned}$$

while $\bar{x}(k) = x(k) - x^0(k)$ *and* $\bar{u}(k) = u(k) - u^0(k)$ *denote variations from the nominal trajectory and* $\mathbf{x}^0 = \left\{x^0(i), x^0(i+1), \dots, x^0(i+N)\right\}$ *is calculated by recursively applying Eq.* (4).

3 MODEL-PREDICTIVE CONTROLLER FOR TRAJECTORY TRACKING AND SET-POINT REGULATION

In this section, error dynamics will be derived based on which a finite-horizon optimal control problem will be formulated. It will be shown that this formulation ensures stability of the closed-loop error dynamics under the receding-horizon controller.

3.1 Transformation to Stabilization

Considering the definition of the state vector in Proposition 2.1, the desired state can be defined as

$$x_r(k) \triangleq \Big[r^T(k) \;\; r^T(k-1) \;\; \dots \;\; r^T(k-n_a) \,\vdots\, u_r^T(k-1) \;\; u_r^T(k-2) \;\; \dots \;\; u_r^T(k-n_{bu}) \Big]. \tag{11}$$

The reference input, $u_r(k)$, satisfies

$$r(k+1) = f_1(k, x_r(k), u_r(k)) \tag{12}$$

where $f_1(k, x_r(k), u_r(k))$ are the first p elements of the vector function $f(k, x_r(k), u_r(k))$. It is assumed that a perfect tracking solution exists and thus Eq. (12) can be solved (numerically) for $m \geq p$. When this assumption is not satisfied, $r(k)$ is often replaced with a reachable target that is close to it [24]. Such trajectory planning considerations are outside the scope of this paper.

Defining the tracking error as $\tilde{x}(k) = x(k) - x_r(k)$, its dynamics can be described as

$$\begin{aligned} \tilde{x}(k+1) &= \tilde{f}(k, \tilde{x}(k), \tilde{u}(k)) \\ \tilde{x}(k) &= x(k) - x_r(k) \\ \tilde{u}(k) &= u(k) - u_r(k) \end{aligned} \tag{13}$$

where $\tilde{f}(k, \tilde{x}(k), \tilde{u}(k)) \triangleq f(x_r(k) + \tilde{x}(k), u_r(k) + \tilde{u}(k)) - x_r(k+1)$. The goal is now to drive $\tilde{x}$ to the origin.

3.2 Stabilizing Model-Predictive Controller

An attractive proposition for controlling the system (13) is to solve an optimal control problem at each time step k, and apply the first element of the optimized control sequence. Defining

$$J_N(\tilde{x}(k), \tilde{\mathbf{u}}(k)) = \sum_{i=k}^{k+N-1} \ell(\tilde{x}(i), \tilde{u}(i)) + V_f(\tilde{x}(k+N)) \tag{14}$$

where $\ell(\tilde{x}(i), \tilde{u}(i)) = \tilde{x}^T(i) Q \tilde{x}(i) + \tilde{u}^T(i) R \tilde{u}(i)$ and $V_f(\tilde{x}(k+N)) = \tilde{x}^T(k+N) Q_f \tilde{x}(k+N)$, a finite-horizon control problem can be postulated as

$$\begin{aligned} \min_{\tilde{\mathbf{u}}(\mathbf{k})} \quad & J_N(\tilde{x}(k), \tilde{\mathbf{u}}(k)) \\ \text{subject to:} \quad & \tilde{x}(k+1) = \tilde{f}(k, \tilde{x}(k), \tilde{u}(k)) \\ & \tilde{x}(0) = x(0) - x_r(0) \\ & \tilde{x}(k+N) \in \tilde{\mathbb{X}}_f \end{aligned} \tag{NL-OCP}$$

where $R = R^T > 0$, $Q = Q^T > 0$ and $Q_f = Q_f^T > 0$ are matrices of appropriate dimension and $\tilde{\mathbb{X}}_f$ is a neighborhood of the origin. Such a procedure is referred to as a *receding-horizon* control policy [26] and will be employed to control (13).

It is well-known that a receding-horizon strategy does not guarantee stability of the closed-loop system [24]. In MPC literature, a number of stabilizing modifications have been proposed that, under some additional assumptions on the cost function, guarantee the stability of the closed loop system. A common class of stabilizing constraints are where $\tilde{\mathbb{X}}_f$ is some suitable neighborhood of the origin [27]. A simple choice for the constraint set could be

$$\tilde{x}(k+N) = \{0\} = \tilde{\mathbb{X}}_f \tag{15}$$

which will be used to formulate an optimal control problem in this paper. The MPC algorithm can now be stated as follows.

Algorithm 3.1. *(Receding-horizon trajectory tracking).*

1. *Initialize. Set $k = 0$ and initialize $x_r(0) = x(0)$. Recursively solve Eq.* (12) *for $u_r(i)$ and construct $x_r(i+1)$ $\forall i = 0, 1, \dots, N-1$.*
2. *Solve* (NL-OCP) *to yield $\tilde{\mathbf{u}}^*(k)$*
3. *Apply $\tilde{u}(k) = \tilde{u}^*(k|k)$*
4. *$k = k+1$*
5. *Solve Eq.* (12) *for $u_r(k+N-1)$*
6. *Use $\tilde{\mathbf{u}}^0 = \{\tilde{u}^*(k|k-1), \tilde{u}^*(k+1|k-1), \dots, \tilde{u}^*(k+N-2|k-1), 0\}$ as a warm start for the optimization to yield $\tilde{\mathbf{u}}^*(k)$. Go to Step 2* □

The key to the stability of $\tilde{x} = 0$ under Algorithm 3.1 is the existence of, and ability to solve (NL-OCP), in which case, the following classical result from MPC literature can be invoked.

Theorem 3.1. *(Stability under end point equilibrium constraint). Let the receding-horizon controller be defined as in Algorithm 3.1, where the optimal control problem solved at each time step is* (NL-OCP) *with $\tilde{\mathbb{X}}_f = \{0\}$. Let the input be defined as $\tilde{u}(k) = \tilde{u}^*(k|k)$. The point $\tilde{x} = 0$ is asymptotically stable for the closed-loop system.*

Proof. *See Theorem 5.2 of [28]*

Remark 3.2. *While the optimization is conducted using the variable $\tilde{u}(k)$, the actual input to the system is $u(k) = \tilde{u}^*(k|k) +$*

$u_r(k)$. *An important step in the trajectory tracking problem is the existence and solution for the* sequence $u_r(i)\,\forall i = 0,1,\ldots,k+N-1$ *that maintains the system response on the desired trajectory. This sequence can be solved for by using a numerical root finding procedure or an auxiliary optimization problem.*

3.3 Solution of the Optimal Control Problem via Sequential Quadratic Programs

To apply Algorithm 3.1, the equality-constrained nonlinear optimal control problem (NL-OCP) must be solved at each time, k. Recent results for solving nonlinear constrained optimal control problems utilizing Sequential Quadratic Programs (SQPs) employ sequential approximations to (NL-OCP) that can be solved efficiently [29, 30] and converge to a local optimum. In this section, an SQP method based on a second order approximation of the cost function and linearization of the constraints is employed to solve (NL-OCP) [29].

The quadratic approximation of (NL-OCP) is obtained as follows. Given a state $\tilde{x}$, and an arbitrary control sequence, $\tilde{\mathbf{u}}^0$ the corresponding state sequence, $\tilde{\mathbf{x}}^0$, can be calculated from Eq. (13). Let us denote $\bar{x} = \tilde{x} - \tilde{x}^0$ and $\bar{u} = \tilde{u} - \tilde{u}^0$ as variations from the nominal trajectory. The cost function is re-expressed as

$$\begin{aligned}\bar{J}_N(\bar{x}(k),\bar{\mathbf{u}}) =& \bar{x}(k+N)^T\hat{Q}_f\bar{x}(k+N) + \bar{x}(k+N)^T\bar{x}^0(k+N) + \\ & \sum_{i=k}^{k+N-1} \bar{x}(i)^T\hat{Q}(i)\bar{x}(i) + \bar{u}(i)^T\hat{R}(i)\bar{u}(i) \\ & + \sum_{i=k}^{k+N-1} \bar{x}(i)^T\bar{x}^0(i) + \bar{u}(i)^T\bar{u}^0(i) \end{aligned} \tag{16}$$

where

$$\begin{aligned}\hat{Q}_f &\triangleq \frac{\partial^2 V_f}{\partial \tilde{x}^2} = Q_f \\ \hat{Q}(i) &\triangleq \frac{\partial^2 \ell}{\partial \tilde{x}^2} = Q \\ \hat{R}(i) &\triangleq \frac{\partial^2 \ell}{\partial \tilde{u}^2} = R \\ \bar{x}^0(i) &\triangleq \frac{\partial \ell}{\partial \tilde{x}}^T = Q\tilde{x}^0(i) \\ \bar{x}^0(k+N) &\triangleq \frac{\partial V_f}{\partial \tilde{x}}^T = Q_f\tilde{x}^0(k+N) \\ \bar{u}^0(i) &\triangleq \frac{\partial \ell}{\partial \tilde{u}}^T = R\tilde{u}^0(i). \end{aligned} \tag{17}$$

The dynamics and constraints are linearized as

$$\bar{x}(k+1) = A(k)\bar{x}(k) + B(k)\bar{u}(k) \tag{18}$$

$$\bar{x}(k) = 0 \tag{19}$$

$$\bar{x}(k+N) + x^0(k+N) - x_r(k+N) = 0. \tag{20}$$

where Eq. (18) is the linearization of Eq. (13).

Thus, the quadratic re-expression of (NL-OCP) with the stabilizing constraint (15) can be stated.

$$\begin{aligned} \min_{\bar{\mathbf{u}}} \quad & \bar{J}_N(\bar{x}(k),\bar{\mathbf{u}}) \\ \text{subject to:} \quad & \bar{x}(k+1) = A(k)\bar{x}(k) + B(k)\bar{u}(k) \\ & \bar{x}(0) = 0 \\ & \bar{x}(k+N) + x^0(k+N) - x_r(k+N) = 0. \end{aligned} \tag{LQ-OCP}$$

The SQP procedure penalizes deviations from the end constraint using a non-differentiable penalty function

$$\begin{aligned} M(\tilde{\mathbf{x}},\tilde{\mathbf{u}}) &= J_N(\tilde{x}(k),\tilde{\mathbf{u}}) + \rho L(\tilde{x}(N)) \\ L(\tilde{x}(N)) &= \max|\tilde{x}(N)| \end{aligned}$$

with $\rho > 0$. The constrained SQP method from [29] is summarized below.

Algorithm 3.2. *(Solution of* (NL-OCP) *via Constrained SQP).*

1. *Set iteration counter $n = 0$ and let $\tilde{\mathbf{u}}_n$ be an arbitrary input sequence. Initialize penalty term, $\rho_0 > 0$, the maximum number of iterations $n_{\max}$ and the tolerance on the control step size, δ.*
2. *Compute $\tilde{\mathbf{x}}_n$ using $\tilde{\mathbf{u}}_n$ and Eq.* (13)
3. *Formulate linear-quadratic subproblem* (LQ-OCP)
4. *Solve* (LQ-OCP) *and let $\bar{\mathbf{u}}^*$ be the solution. Set $\rho_{n+1} = \max\{\rho_n, \sum_{i=1}^n \nu_i(N)\}$ where $\nu_i(N)$ is the i^{th} element of the Lagrange multiplier vector for the end constraint for the linearized problem.*
5. *Let $\bar{\mathbf{u}}^*$ be a search direction and compute $\tilde{\mathbf{u}}_{n+1} = \tilde{\mathbf{u}}_n + \alpha\bar{\mathbf{u}}^*$ where $\alpha = \arg\min M(\tilde{\mathbf{x}},\tilde{\mathbf{u}}_n + \alpha\bar{\mathbf{u}}^*)$ for $0 < \alpha \le 1$.*
6. *Check termination condition. If $n > n_{\max}$ or if $\|\bar{u}\| < \delta$ exit. Otherwise, set $n = n+1$ and return to Step 2.* □

This procedure converges to a local optimum of (NL-OCP) if and only if the system is locally N-step controllable[1] [29]. Since the optimality is not global, Theorem 3.1 cannot be invoked to ensure stability of the closed-loop system. In addition, satisfaction of the end point constraint only occurs as $n \to \infty$. These issues will be addressed in the next section.

4 SUBOPTIMAL REGULATION

In the case of regulation to a set point (with N-step constant disturbances), the difficulties associated with end point constraint satisfaction and sub-optimality can be overcome through

[1] implying that at each n, (LQ-OCP) is feasible

the use of a *dual-mode* controller [22]. The notion is to design a locally stabilizing control law for the reference point, $\tilde{x} = 0$ and characterize an invariant set under this control law, which can be used as the terminal constraint set, $\tilde{\mathbb{X}}_f$ in (15). When the state is outside this set, a sequence that satisfies the constraints for (NL-OCP) is pursued via the SQP optimization procedure. Once the state is inside $\tilde{\mathbb{X}}_f$, the locally stabilizing control law can be employed. In order to design this local controller, the system dynamics near the reference point are linearized, for which a locally stabilizing controller of the form $\tilde{u} = K\tilde{x}$ will be pursued such that $\tilde{\mathbb{X}}_f$ invariant.

The key is to characterize a terminal constraint set $\tilde{\mathbb{X}}_f = W$ that is a convex, compact, and *positively invariant* under the local controller. Once W is characterized, the following suboptimal controller can be employed.

Algorithm 4.1. *(Suboptimal receding-horizon control [22]).*

1. *At time $k = 0$, if $\tilde{x}(0) \in W$ set $\tilde{u}(k) = K\tilde{x}$. Otherwise, utilize the SQP procedure to produce a control sequence, $\tilde{\mathbf{u}}(0)$, that satisfies the initial condition, system dynamics and $\tilde{x}(k+N) \in W$. Set $\tilde{u}(0) = \tilde{u}(0|0)$.*
2. *At time $k > 0$, if $\tilde{x}(k) \in W$ set $\tilde{u}(k) = K\tilde{x}(k)$. Otherwise, use*

$$\mathbf{u}_0 = \{\tilde{u}(k|k-1), \tilde{u}(k+1|k-1), \ldots, \tilde{u}(k+N-2|k-1), K\tilde{x}(k+N-1)\}$$

   as a warm start for the SQP procedure. Perform N_{iter} iterations of the SQP procedure to yield an improved input sequence.
3. *Apply $\tilde{u}(k) = \tilde{u}(k|k)$*
4. *Set $k = k+1$*
5. *Solve Eq. (12) for $u_r(k+N-1)$ and go to Step 2.* □

In this algorithm, the SQP procedure continues until $\tilde{x}(k+N) \in W$ which is a compact set containing $\tilde{x} = 0$. Since the SQP procedure yields $\tilde{x}(k+N) \to 0$ as the number of iterations tends to infinity, there will be a (finite) number of iterations before $\tilde{x}(k+N)$ enters W. It remains to characterize W for the case of regulation.

Let us pursue a W of the form $W = \left\{\tilde{x}\,|\,\tilde{x}^T P\tilde{x} \le \varepsilon\right\}$ where P and ε will be determined such that W is positively invariant under the local linear controller. The closed-loop system for $\tilde{x} \in W$ is $\tilde{x}(k+1) = A_{cl}\tilde{x}(k) + e(\tilde{x})$ where $e(\tilde{x}) \triangleq \tilde{f}(\tilde{x}, K\tilde{x}) - A_{cl}\tilde{x}$ and $A_{cl} = A + BK$. Since A_{cl} is stable, there exists positive definite $P = P^T$ for any positive definite $\bar{Q} = \bar{Q}^T$ such that $A_{cl}^T P A_{cl} - P = -\bar{Q}$.

In the sequel, the following will be required.

Lemma 4.1. *(Cost Reduction and positive invariance of W).* *Define $W = \left\{\tilde{x}\,|\,\tilde{x}^T P\tilde{x} \le \varepsilon\right\}$*

$$\begin{aligned} c_1 &\triangleq \lambda_{\max}(P) \\ c_2 &\triangleq \|PA_{cl}\| \\ c_3 &\triangleq \lambda_{\min}(\bar{Q}) \\ \gamma &< \frac{-c_2 + \sqrt{c_2^2 + (\mu-1)c_3 c_1}}{c_1} \end{aligned}$$

where $\mu > 1$ is otherwise arbitrary, $\lambda_{\max}(\cdot)$ and $\lambda_{\min}(\cdot)$ denote the maximum and minimum eigenvalues and $P = P^T > 0$, $\bar{Q} = \bar{Q}^T > 0$ satisfy $A_{cl}^T P A_{cl} - P = -\mu\bar{Q}$. If, for all $\tilde{x} \in W$, $\tilde{u} = K\tilde{x}$ and the linearization error can be bounded as

$$\|e(\tilde{x})\| \le \gamma\|\tilde{x}\| \tag{21}$$

then $\tilde{x} \in W \implies \tilde{f}(\tilde{x}, K\tilde{x}) \in W$, which means that W is invariant under the linear control law. Furthermore,

$$\Delta V < -\tilde{x}(k)^T \bar{Q}\tilde{x}(k) \quad \forall \tilde{x} \in W. \tag{22}$$

Proof. Consider the Lyapunov-like function $V(\tilde{x}) = \tilde{x}^T P\tilde{x}$ and define $\Delta V = V(\tilde{x}(k+1)) - V(\tilde{x}(k))$. Then

$$\begin{aligned} \Delta V &= (A_{cl}\tilde{x} + e(\tilde{x}))^T P (A_{cl}\tilde{x} + e(\tilde{x})) - \tilde{x}(k)^T P\tilde{x}(k) \\ &= \tilde{x}^T \left(A_{cl}^T P A_{cl} - P\right)\tilde{x} + 2\tilde{x}^T A_{cl}^T P e(\tilde{x}) + e(\tilde{x})^T P e(\tilde{x}) \end{aligned}$$

To ensure that $\Delta V < -\tilde{x}(k)^T \bar{Q}\tilde{x}(k)$ we require that

$$\begin{aligned} \tilde{x}^T \left(A_{cl}^T P A_{cl} - P + \bar{Q}\right)\tilde{x} + 2\tilde{x}^T A_{cl}^T P e(\tilde{x}) + e(\tilde{x})^T P e(\tilde{x}) < 0 \\ \tilde{x}^T (1-\mu)\bar{Q}\tilde{x} + 2\tilde{x}^T A_{cl}^T P e(\tilde{x}) + e(\tilde{x})^T P e(\tilde{x}) < 0 \end{aligned}$$

Employing the error bound $\|e(\tilde{x})\| \le \gamma\|\tilde{x}\|$, the above is true if

$$\left(\lambda_{\max}(P)\gamma^2 + 2\|PA\|\gamma - (\mu-1)\lambda_{\min}(\bar{Q})\right)\|\tilde{x}\|^2 < 0$$

and defining γ as above ensures that $\Delta V = V(\tilde{x}(k+1)) - V(\tilde{x}(k)) < -\tilde{x}(k)^T \bar{Q}\tilde{x}(k)$. Since $W = \{\tilde{x}\,|\,V \le \varepsilon\}$ we have that $\tilde{x} \in W \implies \tilde{f}(\tilde{x}, K\tilde{x}) \in W$.

Lemma 4.2. *(Characterization of W). Using the definitions of P and γ from Lemma 4.1, there exists an $\varepsilon_1 > 0$ such that the linerization error can be bounded as*

$$\|e(\tilde{x})\| \le \gamma\|\tilde{x}\| \quad \forall \|\tilde{x}\| \le \varepsilon_1.$$

Furthermore, selecting $\varepsilon = \lambda_{\max}(P)\varepsilon_1^2$ for $W = \{\tilde{x} \,|\, \tilde{x}^T P \tilde{x} < \varepsilon\}$ implies that W is positive invariant under $\tilde{u} = K\tilde{x}$.

Proof. *see [24] pp. 137.*

The stability of the suboptimal controller can now be established.

Theorem 4.3. *(Stability of the sub-optimal controller). Take $\bar{Q} = (Q + K^T RK)$ and let $Q_f = P$ and let W be characterized by Lemmas 4.2 and 4.1. The sub-optimal controller of Algorithm 4.1 renders the point $\tilde{x} = 0$ asymptotically stable.*

Proof. *The proof largely follows the suboptimal controller idea in [24]. At time k, assume a feasible input sequence $\tilde{\mathbf{u}}(k)$ is found. The cost of this sequence is $J_N(\tilde{x}, \tilde{\mathbf{u}}(k))$. Now, at time $k+1$ the following input sequence is feasible*

$$\mathbf{w} = \{\tilde{u}(k+1|k), \tilde{u}(k+2|k), \ldots, \tilde{u}(k+N-1|k), K\tilde{x}(k+N)\}$$

with the associated cost

$$\begin{aligned} J_N(\tilde{x}(k+1), \mathbf{w}) &= \sum_{i=1}^{N} \ell(\tilde{x}(k+i), \tilde{u}(k+i)) + V_f(\tilde{x}(k+N+1)) \\ &= J_N(\tilde{x}(k), \mathbf{u}(k)) + \ell(\tilde{x}(k+N), \tilde{u}(k+N)) \\ &\quad - \ell(\tilde{x}(k), \tilde{u}(k)) - V_f(\tilde{x}(k+N)) \\ &\quad + V_f(\tilde{x}(k+N+1)) \end{aligned}$$

Since $\tilde{x}(k+N) \in W$, $\tilde{u}(k+N) = K\tilde{x}(k+N)$, if we can guarantee that

$$V_f(\tilde{x}(k+1)) - V_f(\tilde{x}(k)) \leq -\ell(\tilde{x}(k), K\tilde{x}(k)) \quad \forall \tilde{x}(k) \in W \quad (23)$$

it follows that

$$J_N(\tilde{x}(k+1), \mathbf{w}) \leq J_N(\tilde{x}(k), \mathbf{u}(k)) - \ell(\tilde{x}(k), \tilde{u}(k))$$

without any optimization. By taking $V_f(x(k+N)) = \tilde{x}^T(k+N)P\tilde{x}^T(k+N)$, the condition in Eq. (23) is assured. □

Note that the optimal solution is not required here, only that the SQP procedure produces a control sequence that yields a terminal state that is within W.

The following details a method for characterizing W from the local models in Eq. (13).

Proposition 4.4. *(Characterization of W). For the system defined by Eq. (13), an approximate bound on the linearization error is*

$$\begin{aligned} \|e(\tilde{x})\| &\lesssim \sum_{m=1}^{M} \left\| \left(\frac{\partial \nu_m}{\partial x} + \frac{\partial \nu_m}{\partial u} K \right) \right\| \left\| A_m^{cl} \right\| \|\tilde{x}\|^2 \\ &= c\|\tilde{x}\|^2 \\ A_m^{cl} &\triangleq A_m + B_m K \end{aligned}$$

for small $\tilde{x}$. This bound can be used to obtain a bound of the form

$$\|e(\tilde{x})\| \lesssim \gamma \|\tilde{x}\|$$

for $\|\tilde{x}\| < \hat{\varepsilon}_1$. Here γ is obtained as in Lemma 4.1 and $\hat{\varepsilon}_1 = \frac{\gamma}{c}$. This estimation can be used directly in Lemma 4.2 to characterize W.

Proof. *Using the definition of $e(\tilde{x})$ we have*

$$\begin{aligned} e(\tilde{x}) &= \tilde{f}(\tilde{x}, K\tilde{x}) - A_{cl}\tilde{x} \\ &= f(x_r + \tilde{x}, u_r + K\tilde{x}) - f(x_r, u_r) - A_{cl}\tilde{x} \end{aligned}$$

since $x_r = f(x_r, u_r)$. Next, using the system model we have

$$\begin{aligned} e(\tilde{x}) &= \left[\sum_{m=1}^{M} \nu_m(x_r + \tilde{x}, u_r + K\tilde{x}) F_m(x_r + \tilde{x}, u_r + K\tilde{x}) \right. \\ &\quad \left. - \nu_m(x_r, u_r) F_m(x_r, u_r) \right] - A\tilde{x} - BK\tilde{x} \\ &= \left[\sum_{m=1}^{M} \nu_m(x_r + \tilde{x}, u_r + K\tilde{x}) \left[F_m(x_r, u_r) + A_m^{cl}\tilde{x} \right] \right. \\ &\quad \left. - \nu_m(x_r, u_r) F_m(x_r, u_r) \right] - A\tilde{x} - BK\tilde{x} \end{aligned}$$

where the second line uses the definition of F_m in Eq. (6). Defining $\Delta\nu_m \triangleq \nu_m(x_r + \tilde{x}, u_r + K\tilde{x}) - \nu_m(x_r, u_r)$, we have

$$\begin{aligned} e(\tilde{x}) &= \left[\sum_{m=1}^{M} \nu_m(x_r, u_r) A_m^{cl}\tilde{x} + \Delta\nu_m F_m(x_r, u_r) + \right. \\ &\quad \left. \Delta\nu_m A_m^{cl}\tilde{x} \right] - A\tilde{x} - BK\tilde{x}. \end{aligned}$$

Using the definitions of A and B above and noting that $\frac{\partial \nu_m}{\partial x}\tilde{x}$ and $\frac{\partial \nu_m}{\partial u}K\tilde{x}$ are scalars, we have

$$\begin{aligned} e(\tilde{x}) &= \sum_{m=1}^{M} \left[\Delta\nu_m - \frac{\partial \nu_m}{\partial x}\tilde{x} - \frac{\partial \nu_m}{\partial u}K\tilde{x} \right] F_m(x_r, u_r) + \\ &\quad \Delta\nu_m A_m^{cl}\tilde{x}. \end{aligned}$$

For small $\tilde{x}$

$$\Delta\nu_m \approx \frac{\partial \nu_m}{\partial x}\tilde{x} + \frac{\partial \nu_m}{\partial u}K\tilde{x}$$

which can be used to find an approximate upper bound on $\|e(\tilde{x})\|$ as

$$\begin{aligned}\|e(\tilde{x})\| &\approx \left\| \sum_{m=1}^{M} \tilde{x}^T \left[\frac{\partial \nu_m}{\partial x} + \frac{\partial \nu_m}{\partial u}K\right]^T A_m^{cl}\tilde{x} \right\| \\ &\leq \sum_{m=1}^{M} \left\| \left(\frac{\partial \nu_m}{\partial x} + \frac{\partial \nu_m}{\partial u}K\right) \right\| \left\| A_m^{cl} \right\| \|\tilde{x}\|^2 \qquad \square\end{aligned}$$

Please note that the region estimated by the method in Proposition 4.4 is not the largest possible. Determination of a locally stabilizing controller that maximizes the size of this region could be beneficial (fewer SQP iterations). However, such benefits would come at the cost of an additional procedure to determine the controller that maximizes W and such considerations are outside the scope of this paper.

5 APPLICATION TO A CONTINUOUSLY-STIRRED TANK REACTOR

Let us consider a Continuously-Stirred Tank Reactor with the coolant flow rate as the control signal. A diagram of this system can be seen in Fig. 1 [19, 31]. Per [31], the pertinent

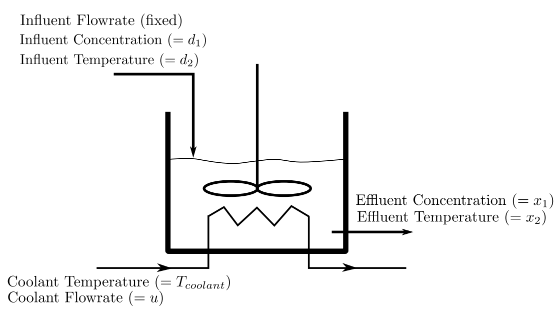

Figure 1. A continuously-stirred tank reactor

dynamic equations are

$$\begin{aligned}\dot{x}_1 &= d_1 - x_1 + a_0 x_1 \exp\left(\frac{-10^4}{x_2}\right) \\ \dot{x}_2 &= d_2 - x_2 + a_1 x_1 \exp\left(\frac{-10^4}{x_2}\right) + \\ &\quad a_3 u \left(1 - \exp\left(\frac{-a_2}{u}\right)\right)(T_{coolant} - x_2)\end{aligned}$$

and it is obvious that the system is nonlinear in the control. The nominal system parameters are given in [31]. The output of the system is considered to be the effluent concentration, x_1. A sampling time of 0.1 was used and a local ARX model was built using an algorithm similar to [32], but with continuously differentiable local activation functions

$$\begin{aligned}\tilde{\nu}_i(w(k)) &= \frac{h_i(w(k))}{\sum_{m=1}^{M} h_m(w(k))} \\ h_m(w(k)) &= \exp\left[\frac{-\|w(k) - \xi_j\|^2}{2\sigma^2}\right]\end{aligned}$$

where ξ_j denotes the j^{th} local region center. The model orders used in the training were $n_a = 2$, $n_{bu} = 2$, $n_{bd} = [1,1]$ and the scheduling vector was $w(k) = [x_1(k), x_1(k-1), u(k), u(k-1)]^T$. Both selections were motivated by physical considerations, recognizing that the nonlinearity only depends on the input and x_2. Note that in this paper no restriction has been made as to where the local models come from. For example, one could generate local affine state-space models from off-equilibrium linearizations and apply the MPC methodology of this paper.

The algorithm of Section 4 was applied to regulate the concentration at a desired level with the control horizon set to to $N = 5$. The reference was piecewise constant, alternating between $r = 0.12$ and $r = 0.10$ in 16 minute intervals.The results can be seen in Fig. 2. If, instead, the reference is generated by

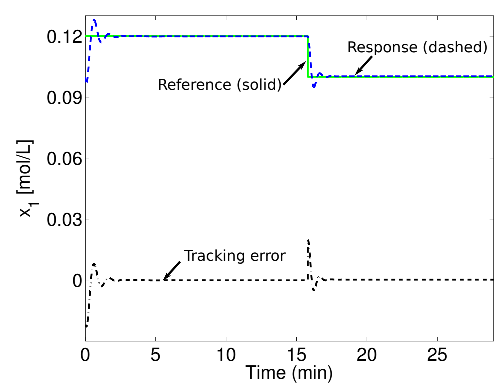

Figure 2. Suboptimal regulation applied to the CSTR system

the model

$$\begin{aligned}r(k+1) &= 0.9r(k) + 0.1\upsilon(k) \\ \upsilon(k) &= 0.095 + 0.025\sin\left(\frac{1}{3}k\Delta t\right),\end{aligned}$$

the problem becomes a tracking problem. The MPC control algorithm may still be applied, but the termination criterion for the

SQP procedure is now simply set *ad hoc*. Despite this, the algorithm was found to perform well as long as the system operated in regions which the local ARX model approximated the actual system well. The simulation result can be seen in Fig. 3.

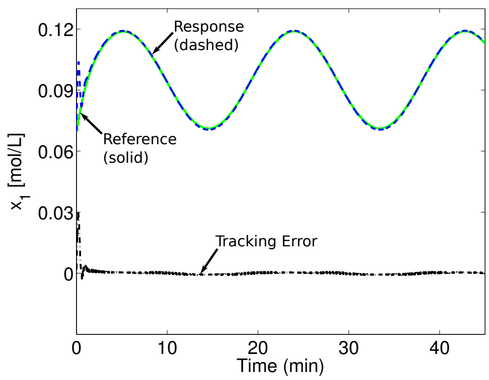

Figure 3. Tracking performance of the MPC controller

6 CONCLUSIONS

In this paper, a model-predictive control scheme for nonlinear systems described by local ARX models was presented. The model structure is permitted to have nonlinearities in the state and control input and a control algorithm was formulated that guarantees stability. For the case of set point regulation, a stabilizing control algorithm modification was established that does not require achievement of the optimal solution to the finite-horizon nonlinear optimal control problem and enables finite termination of the iterative procedure from which the control signal is determined. Future work will focus on extension of the suboptimal policy to tracking problems and robust formulations of the MPC methods introduced in this paper.

REFERENCES

[1] Wang, H., Tanaka, K., and Griffin, M., 1995. "Parallel distributed compensation of nonlinear systems by Takagi-Sugeno fuzzy model". *Proceedings of 1995 IEEE International Conference on Fuzzy Systems,* **2**, pp. 531–538.

[2] Tuan, H., Apkarian, P., Narikiyo, T., and Kanota, M., 2004. "New fuzzy control model and dynamic output feedback parallel distributed compensation". *IEEE Transactions on Fuzzy Systems,* **12**(1), pp. 13–21.

[3] Li, J., Wang, H., Niemann, D., and Tanaka, K., 2000. "Dynamic parallel distributed compensation for TakagiSugeno fuzzy systems: an LMI approach". *Information sciences,* **123**(3-4), Apr., pp. 201–221.

[4] Fantuzzi, C., and Rovatti, R., 1996. "On the approximation capabilities of the homogeneous Takagi-Sugeno model". *Fuzzy Systems, 1996., Proceedings of the Fifth IEEE International Conference on,* **2**, pp. 1067–1072.

[5] Rodrigues, L., and How, J., 2003. "Observer-based control of piecewise-affine systems". *International Journal of Control,* **76**(5), Jan., pp. 459–477.

[6] Samadi, B., and Rodrigues, L., 2008. "Extension of local linear controllers to global piecewise affine controllers for uncertain non-linear systems". *International Journal of Systems Science,* **39**(9), Sept., pp. 867–879.

[7] Mignone, D., Ferrari-trecate, G., and Morari, M., 2000. "Stability and stabilization of piecewise affine and hybrid systems: An LMI approach". *IEEE Conference on Decision and Control,* **1**, pp. 504–509.

[8] Feng, G., 2003. "Output Feedback Controller Design of Piecewise Discrete Time Linear Systems". In The Fourth International Conference on Control and Automation (ICCCA'03), no. June, pp. 43–47.

[9] Gao, Q., Zeng, X.-J., Feng, G., Wang, Y., and Qiu, J., 2012. "T-S-Fuzzy-Model-Based Approximation and Controller Design for General Nonlinear Systems". *IEEE Transactions on Systems, Man and Cybernetics, Part B: Cybernetics,* **42**(4), Mar., pp. 1143–1154.

[10] Roubos, J., Mollov, S., Babuska, R., and Verbruggen, H., 1999. "Fuzzy model-based predictive control using Takagi-Sugeno models". *International Journal of Approximate Reasoning,* **22**, pp. 3–30.

[11] Abonyi, J., Nagy, L., and Szeifert, F., 2001. "Fuzzy model-based predictive control by instantaneous linearization". *Fuzzy Sets and Systems,* **120**(1), May, pp. 109–122.

[12] Li, N., Li, S., and Xi, Y., 2004. "Multi-model predictive control based on the Takagi-Sugeno fuzzy models: a case study". *Information Sciences,* **165**(3-4), Oct., pp. 247–263.

[13] Hadjili, M., and Wertz, V., 1999. "Generalized predictive control using Takagi-Sugeno fuzzy models". In Proceedings of the 1999 IEEE International Symposium on Intelligent Control/Intelligent Systems and Semiotics, Ieee, pp. 405–410.

[14] Fischer, M., Nelles, O., and Isermann, R., 1998. "Predictive control based on local linear fuzzy models". *International Journal of Systems Science,* **29**(7), pp. 679–697.

[15] Dziekan, L., Witczak, M., and Korbicz, J., 2011. "A predictive fault-tolerant control scheme for Takagi-Sugeno fuzzy systems". In Proceedings of the 18th IFAC World Congress, pp. 4684–4689.

[16] Zhang, T., Feng, G., and Zeng, X., 2009. "Output tracking of constrained nonlinear processes with offset-free input-to-state stable fuzzy predictive control". *Automatica,* **45**(4), Apr., pp. 900–909.

[17] S{\o}rensen, P., N{\o}rgaard, M., Ravn, O., and Poulsen, N., 1999. "Implementation of neural network based nonlinear predictive control". *Neurocomputing,* **28**(1-3), Oct., pp. 37–51.

[18] Lu, C., and Tsai, C., 2007. "Generalized predictive control using recurrent fuzzy neural networks for industrial processes". *Journal of Process Control,* **17**(1), Jan., pp. 83–92.

[19] Nikravesh, M., Farell, A., and Stanford, T., 2000. "Con-

trol of nonisothermal CSTR with time varying parameters via dynamic neural network control (DNNC)". *Chemical Engineering Journal,* **76**(1), Jan., pp. 1–16.
[20] Johansen, T., 1996. "Stability, robustness, and performance of fuzzy model based control". *Proceedings of the 35th Conference on Decision and Control,* **1**, pp. 604–609.
[21] Liu, J., Djurdjanovic, D., Marko, K., and Ni, J., 2009. "Growing Structure Multiple Model Systems for Anomaly Detection and Fault Diagnosis". *Journal of Dynamic Systems, Measurement, and Control,* **131**(5), Sept., pp. 051001–13.
[22] Scokaert, P., Mayne, D., and Rawlings, J., 1999. "Suboptimal model predictive control (feasibility implies stability)". *IEEE Transactions on Automatic Control,* **44**(3), pp. 648–654.
[23] Nelles, O., 2001. *Nonlinear System Identification*. Springer-Verlag, Berlin Heidelberg.
[24] Rawlings, J. B., and Mayne, D. Q., 2009. *Model Predictive Control: Theory and Design*. Nob Hill Publishing, Madison, Wisconsin.
[25] Camacho, E. F., and Bordons, C., 1999. *Model Predictive Control*. Springer-Verlag, London.
[26] Grüne, L., and Pannek, J., 2011. *Nonlinear Model Predictive Control: Theory and Algorithms*. Springer-Verlag, London.
[27] Mayne, D., Rawlings, J., Rao, C., and Scokaert, P., 2000. "Constrained model predictive control : Stability and optimality". *Automatica,* **36**, pp. 789–814.
[28] Keerthi, S., and Gilbert, E., 1988. "Optimal infinite-horizon feedback laws for a general class of constrained discrete-time systems: Stability and moving-horizon approximations". *Journal of Optimization Theory and Applications,* **57**(2), pp. 265–293.
[29] Rodriguez, L. A., and Sideris, A., 2011. "A sequential linear quadratic approach for constrained nonlinear optimal control". In American Control Conference (ACC), pp. 1470–1475.
[30] Sideris, A., and Bobrow, J., 2005. "An efficient sequential linear quadratic algorithm for solving nonlinear optimal control problems". *IEEE Transactions on Automatic Control,* **50**(12), pp. 2043–2047.
[31] Ge, S., Zhang, J., and Lee, T., 2004. "Adaptive MNN control for a class of non-affine NARMAX systems with disturbances". *Systems & Control Letters,* **53**(1), Sept., pp. 1–12.
[32] Cholette, M., and Djurdjanovic, D., 2012. "Precedent-Free Fault Isolation in a Diesel Engine Exhaust Gas Recirculation System". *Journal of Dynamic Systems, Measurement, and Control,* **134**(3), p. 031007.

Appendix A: Matrices for State-Space Formulation

$$\hat{A}_i = \left[\begin{array}{c:c:c} A_{11}^{(i)} & A_{12}^{(i)} & A_{13}^{(i)} \\ \hdashline 0 & A_{22}^{(i)} & 0 \\ \hdashline 0 & 0 & A_{33}^{(i)} \end{array}\right]$$

$$A_{11}^{(i)} = \begin{bmatrix} q_0 & q_1 & \dots & q_{n_a-1} & q_{n_a} \\ I_p & 0 & \dots & 0 & 0 \\ 0 & I_p & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & I_p & 0 \end{bmatrix}$$

$$A_{12}^{(i)} = \begin{bmatrix} q_{n_a+2} & \dots & q_{n_a+n_b-1} & q_{n_a+n_{bu}} \\ 0 & \dots & 0 & 0 \\ 0 & \dots & 0 & 0 \\ \vdots & \ddots & \vdots & \vdots \\ 0 & \dots & 0 & 0 \end{bmatrix}$$

$$A_{13}^{(i)} = \begin{bmatrix} q_{n_a+n_{bu}+2} & \dots & q_{n_a+n_{bu}+n_{du}-1} & q_{n_a+n_{bu}+n_{du}} \\ 0 & \dots & 0 & 0 \\ 0 & \dots & 0 & 0 \\ \vdots & \ddots & \vdots & \vdots \\ 0 & \dots & 0 & 0 \end{bmatrix}$$

$$A_{22}^{(i)} = \begin{bmatrix} 0 & \dots & 0 & 0 \\ I_m & \dots & 0 & 0 \\ \vdots & \ddots & \vdots & \vdots \\ 0 & \dots & I_m & 0 \end{bmatrix} \quad A_{33}^{(i)} = \begin{bmatrix} 0 & \dots & 0 & 0 \\ I_q & \dots & 0 & 0 \\ \vdots & \ddots & \vdots & \vdots \\ 0 & \dots & I_q & 0 \end{bmatrix}$$

$$\hat{B}_i = \left[\begin{array}{c} B_1^{(i)} \\ \hdashline B_2^{(i)} \\ \hdashline 0 \end{array}\right] = \left[\begin{array}{cccc:cccc:ccc} q_{n_a+1}^T & 0 & \dots & 0 & I_m & 0 & \dots & 0 & 0 & \dots & 0 \end{array}\right]^T$$

$$\hat{G}_i = \left[\begin{array}{c} G_1^{(i)} \\ \hdashline 0 \\ \hdashline G_2^{(i)} \end{array}\right] = \left[\begin{array}{cccc:ccc:cccc} q_{n_a+n_{bu}+1}^T & 0 & \dots & 0 & 0 & \dots & 0 & I_q & 0 & \dots & 0 \end{array}\right]^T$$

$$\hat{b}_i = \left[\begin{array}{c} b_1^{(i)} \\ \hdashline 0 \\ \hdashline 0 \end{array}\right] = \left[\begin{array}{ccccc:cccc:cccc} b_i^T & 0 & 0 & \dots & 0 & 0 & 0 & \dots & 0 & 0 & 0 & \dots & 0 \end{array}\right]^T$$

Here I_p,I_m and I_q are identity matrices and q_j are sub-matrices of Q_i. For $j \leq n_a$, $q_j \in \mathbb{R}^{p\text{x}p}$ for $n_a < j \leq n_{bu}$, $q_j \in \mathbb{R}^{p\text{x}m}$ and for $n_a + n_{bu} < j \leq n_a + n_{bu} + n_{bd}$, $q_j \in \mathbb{R}^{p\text{x}q}$.