Integration of Real-time Optimization & and Model Predictive Control ⋆

V. Adetola * **M. Guay** *

* *Department of Chemical Engineering, Queen's University, Kingston, Ontario, Canada (e-mail: martin.guay@chee.queensu.ca)*

Abstract: This paper proposes a controller design approach that integrates RTO and MPC for the control of constrained uncertain nonlinear systems. Assuming that the economic function is a known function of constrained system's states, parameterized by unknown parameters and time-varying, the controller design objective is to simultaneously identify and regulate the system to the optimal operating point. The approach relies on a novel set-based parameter estimation routine and a robust model predictive controller that takes into the effect of parameter estimation errors. A simulation example is used to demonstrate the effectiveness of the design technique.

Keywords: Adaptive control, Real-time optimization, Model predictive control

1. INTRODUCTION

In this paper, we provide a formal design technique that integrates RTO and MPC for constrained uncertain nonlinear systems. The framework considered assumes the economic function is a known function of constrained system's states, parameterized by unknown parameters. The objective and constraint functions may explicitly depend on time, which means that our proposed method is applicable to both dynamic and steady state economic optimization. The control objective is to simultaneously identify and regulate the system to the operating point that optimizes the economic function. The control input may also be required to satisfy some constraints.

The method proposed solves the control and optimization problem at the same frequency. This eliminates the ensuing interval of "no-feedback" that occurs between economic optimization and thereby improving disturbance attenuation. The RTO layer is tackled via a computational efficient approach. The constrained economic optimization problem is converted to an unconstrained problem and Newton based optimization method is used to develop an update law for the optimum value. The integrated design distinguishes between the extremum seeking and the adaptive tracking of the reference trajectory.

While many advances have been made in nonlinear systems for the stabilization of one fixed operating point, few attempts have been made to address the stabilization problem for time-varying or non-fixed setpoints. In Magni (2002), a stabilizing nonlinear MPC algorithm was developed for asymptotically constant reference signals. By selecting a prediction horizon that is longer than the time the reference setpoint is assumed to have converged, the constant pre-programmed value is used to design the stabilizing controller parameters, *i.e*, the terminal stability constraint $\mathbb{X}_f$ and terminal penalty W. The result is limited to reference signals that converge to *a-priori* known constant setpoint. The method proposed in Findeisen et al. (2000), combines a pseudo-linearization technique with a nonlinear MPC strategy to stabilize a family of (known and constant) setpoints. While the method provides a possible solution for tracking changing setpoints, such pseudo-linearization transformation and feedback is in general difficult to obtain and involve cumbersome computation.

⋆ The authors would like to acknowledge the financial support of the Natural Sciences and Engineering Research Council of Canada.

2. PROBLEM DESCRIPTION

Consider a constrained optimization problem of the form

$$\min_{x\in\mathbb{R}^{n_x}} \; p(t,x,\theta) \tag{1a}$$

$$s.t. \quad c_j(x) \le 0 \qquad j = 1\ldots m_c \tag{1b}$$

with θ representing unknown parameters, assumed to be uniquely identifiable and lie within an initially known convex set $\Theta^0 \triangleq B(\theta^0, z_\theta^0)$. The functions p and c_j are assumed to be C^2 in all of their arguments (with locally Lipschitz second derivatives), uniformly for $t \in [0,\infty)$. The constraint $c_j \le 0$ must be satisfied along the system's state trajectory $x(t)$.

Assumption 1. The following assumptions are made about (1).

(1) There exists $\varepsilon_0 > 0$ such that $\frac{\partial^2 p}{\partial x^2} \ge \varepsilon_0 I$ and $\frac{\partial^2 c}{\partial x^2} \ge 0$ for all $(t,x,\theta) \in (\mathbb{R}^+ \times \mathbb{R}^{n_x} \times \Theta^\epsilon)$, where Θ^ϵ is an ϵ neighborhood of Θ.
(2) The feasible set
$$\mathbb{X} = \left\{x \in \mathbb{R}^{n_x} \mid \max_j c_j(x) \le 0\right\},$$
has a nonempty interior.

Assumption 1 states that the cost surface is strictly convex in x and $\mathbb{X}$ is a non-empty convex set. Standard nonlinear optimization results guarantee the existence of a unique minimizer $x^*(t,x,\theta) \in \mathbb{X}$ to problem 1. In the case of non-convex cost surface, only local attraction to an extremum could be guaranteed. The control objective is to stabilize the nonlinear system

$$\dot{x} = f(x,\xi,u) + g(x,\xi,u)\theta \triangleq \mathcal{F}(x,\xi,u,\theta) \quad (2a)$$
$$\dot{\xi} = f_\xi(x,\xi) \quad (2b)$$

to the optimum operating point or trajectory given by the solution of (1) while obeying the input constraint $u \in \mathbb{U} \in \mathbb{R}^{n_u}$ in addition to the state constraint $x \in \mathbb{X} \in \mathbb{R}^{n_x}$. The dynamics of the state ξ is assumed to satisfy the following input to state stability condition with respect to x.

Assumption 2. If x is bounded by a compact set $B_x \subseteq \mathbb{X}$, then there exists a compact set $B_\xi \subseteq \mathbb{R}^{n_\xi}$ such that $\xi \in B_\xi$ is positively invariant under 2.

3. EXTREMUM SEEKING SETPOINT DESIGN

3.1 Finite-time Parameter Identification

Let $\hat{x}$ denote the state predictor for (2), the dynamics of the state predictor is designed as

$$\dot{\hat{x}} = f(x,\xi,u) + g(x,\xi,u)\hat{\theta} + k_w(t)e + w\dot{\hat{\theta}}, \quad (3)$$

where $\hat{\theta}$ is a parameter estimate generated via any update law $\dot{\hat{\theta}}$, $k_w > 0$ is a design matrix, $e = x - \hat{x}$ is the prediction error and w is the output of the filter

$$\dot{w} = g(x,\xi,u) - k_w w, \qquad w(t_0) = 0. \quad (4)$$

Denoting the parameter estimation error as $\tilde{\theta} = \theta - \hat{\theta}$, it follows from (2) and (3) that

$$\dot{e} = g(x,\xi,u)\tilde{\theta} - k_w\, e - w\dot{\hat{\theta}}. \quad (5)$$

The use of the filter matrix w in the above development provides direct information about parameter estimation error $\tilde{\theta}$ without requiring a knowledge of the velocity vector $\dot{x}$. This is achieved by defining the auxiliary variable

$$\eta = e - w\tilde{\theta} \quad (6)$$

with η, in view of (4, 5), generated from

$$\dot{\eta} = -k_w\, \eta, \qquad \eta(t_0) = e(t_0). \quad (7)$$

Based on the dynamics (3), (4) and (7), the main result is given by the following theorem.

Theorem 3. Let $Q \in \mathbb{R}^{n_\theta \times n_\theta}$ and $C \in \mathbb{R}^{n_\theta}$ be generated from the following dynamics:

$$\dot{Q} = w^T w, \qquad Q(t_0) = 0 \quad (8a)$$
$$\dot{C} = w^T(w\hat{\theta} + e - \eta), \qquad C(t_0) = 0 \quad (8b)$$

Suppose there exists a time t_c and a constant $c_1 > 0$ such that $Q(t_c)$ is invertible *i.e.*

$$Q(t_c) = \int_{t_0}^{t_c} w^T(\tau)w(\tau)\, d\tau \succ c_1 I, \quad (9)$$

then

$$\theta = Q(t)^{-1}C(t) \qquad \text{for all} \quad t \geq t_c. \quad (10)$$

Proof: The result can be easily shown by noting that

$$Q(t)\,\theta = \int_{t_0}^{t} w^T(\tau)w(\tau)\Big[\hat{\theta}(\tau) + \tilde{\theta}(\tau)\Big]\, d\tau. \quad (11)$$

Using the fact that $w\tilde{\theta} = e - \eta$, it follows from (11) that

$$\theta = Q(t)^{-1}\int_{t_0}^{t} \dot{C}(\tau)\, d\tau = Q(t)^{-1}C(t) \quad (12)$$

and (12) holds for all $t \geq t_c$ since $Q(t) \succeq Q(t_c)$. ∎

The result in theorem 3 is independent of the control u and parameter identifier $\dot{\hat{\theta}}$ structure used for the state prediction (eqn 3). Moreover, the result holds if a nominal estimate θ^0 of the unknown parameter (no parameter adaptation) is employed in the estimation routine. In this case, $\hat{\theta}$ is replaced with θ^0 and the last part of the state predictor (3) is dropped ($\dot{\hat{\theta}} = 0$).

Let

$$\theta^c \triangleq Q(t_c)^{-1}\, C(t_c) \quad (13)$$

The finite-time (FT) identifier is given by

$$\hat{\theta}^c(t) = \begin{cases} \hat{\theta}(t), & \text{if} \ \ t < t_c \\ \theta^c, & \text{if} \ \ t \geq t_c. \end{cases} \quad (14)$$

3.2 Constraint Removal

An interior point barrier function method is used to enforce the inequality constraint. The state constraint is incorporated by augmenting the cost function p as follows:

$$p_a(t,x,\theta) \triangleq p(t,x,\theta) - \frac{1}{\eta_c}\sum_{j=1}^{m_c} \ln(-c_j(x)) \quad (15)$$

with $\eta_c > 0$, a fixed constant. The augmented cost function (15) is strictly convex in x and the unconstrained minimization of p_a therefore has a unique minimizer in int$\{\mathbb{X}\}$ which converges to that of (1) in the limit as $\eta_c \to \infty$ Bertsekas (1995).

3.3 Setpoint Update Law

Let $x_r \in \mathbb{R}^{n_x}$ denote a reference setpoint to be tracked by x and $\hat{\theta}$ denote an estimate of the unknown parameter θ. A setpoint update law $\dot{x}_r$ can be designed based on newton's method, such that $x_r(t)$ converges exponentially to the (unknown) $\hat{\theta}$ dependent optimum value of (15). To this end, consider an optimization Lyapunov function candidate

$$V_r = \tfrac{1}{2}\|\frac{\partial p_a}{\partial x}(t,x_r,\hat{\theta})\|^2 \triangleq \tfrac{1}{2}\|z_r\|^2 \quad (16)$$

For the remainder of this section, omitted arguments of p_a and its derivatives are evaluated at $(t, x_r, \hat{\theta})$. Differentiating (16) yields

$$\dot{V}_r = \frac{\partial p_a}{\partial x}\left(\frac{\partial^2 p_a}{\partial x \partial t} + \frac{\partial^2 p_a}{\partial x^2}\dot{x}_r + \frac{\partial^2 p_a}{\partial x \partial \theta}\dot{\hat{\theta}}\right). \quad (17)$$

Using the update law

$$\dot{x}_r = -\left(\frac{\partial^2 p_a}{\partial x^2}\right)^{-1}\left[\frac{\partial^2 p_a}{\partial x \partial t} + \frac{\partial^2 p_a}{\partial x \partial \theta}\dot{\hat{\theta}} + k_r\frac{\partial p_a^T}{\partial x}\right] \triangleq f_r(t,x_r,\hat{\theta}) \quad (18)$$

with $k_r > 0$ and $r(0) = r_0 \in \text{int}\,\{\mathbb{X}\}$ results in

$$\dot{V}_r \leq -k_r\|z_r\|^2, \quad (19)$$

which implies that the gradient function z_r converges exponentially to the origin.

Lemma 4. Suppose $(\theta, \hat{\theta})$ is bounded, the optimal setpoint $x_r(t)$ generated by (18) is feasible and converges to $x^*_{pa}(\hat{\theta})$, the minimizer of (15) exponentially.

Proof: Feasibility follows from the boundedness of $(\theta, \hat{\theta})$ and Assumption 1.1 while convergence follows from (19) and the fact that z_r is a diffeomorphism. ■

4. ONE-LAYER INTEGRATION APPROACH

Since the true optimal setpoint depends on θ, the actual desired trajectory $x_r^*(t, \theta)$ is not available in advance. However, $x_r(t, \hat{\theta})$ can be generated from the setpoint update law (18) and the corresponding reference input $u_r(x_r)$ can be computed on-line.

Assumption 5. $x_r(t, \hat{\theta})$ is such that there exists $u_r(x_r)$ satisfying

$$0 = f(x_r, u_r, \hat{\theta}) \tag{20}$$

The design objective is to design a model predictive control law such that the true plant state x tracks the reference trajectory $x_r(t,\, \hat{\theta})$. Given the desired time varying trajectory (x_r, u_r), an attractive approach is to transform the tracking problem for a time-invariant system into a regulation problem for an associated time varying control system in terms of the state error $x_e = x - x_r$ and stabilize the $x_e = 0$ state. The formulation requires the MPC controller to drive the tracking error x_e into the terminal set $\mathbb{X}_{e_f}(\tilde{\theta})$ at the end of the horizon. Since the system's dynamics is uncertain, we use the finite-time identifier (34) for online parameter adaptation and incorporate robust features in to the adaptive controller formulation to account for the impact of the parameter estimation error $\tilde{\theta}$ in the design.

4.1 Min-max Adaptive MPC

Feedback min-max robust MPC is employed to provide robustness for the MPC controller during the adaptation phase. The controller maximizes a cost function with respect to θ and minimizes it over feedback control policies κ.

The integrated controller is given as

$$u = \kappa_{mpc}(t, x_e, \hat{\theta}) \triangleq \kappa^*(0, x_e, \hat{\theta}) \tag{21a}$$

$$\kappa^* \triangleq \arg \min_{\kappa(\cdot,\cdot,\cdot,\cdot)} J(t, x_e, \hat{\theta}, \kappa) \tag{21b}$$

where $J(t, x_e, \hat{\theta}, \kappa)$ is the (worst-case) cost associated with the optimal control problem:

$$J(t, x_e, \hat{\theta}, \kappa) \triangleq \max_{\theta \in \Theta} \int_0^T L(\tau, x_e^p, u^p, u_r) d\tau \tag{22a}$$

$$+W(\,\tau, x_e^p(T), \tilde{\theta}^p(T)\,) \tag{22b}$$

$s.t.\;\; \forall \tau \in [0,\, T]$

$$\dot{x}^p = f(x^p, \xi^p, u^p) + g(x^p, \xi^p, u^p)\,\theta,\;\; x^p(0) = x \tag{22c}$$

$$\dot{\xi}^p = f(x^p, \xi^p),\;\; \xi^p(0) = \xi \tag{22d}$$

$$\dot{x}_r^p = f_r(t, x_r, \theta),\;\; x_r^p(0) = x_r \tag{22e}$$

$$x_e^p = x^p - x_r^p \tag{22f}$$

$$\dot{w}^p = \beta(g^T(x^p, \xi^p, u^p) - k_w w^p),\;\; w^p(0) = w \tag{22g}$$

$$\dot{Q}^p = \beta({w^p}^T w^p),\;\; Q^p(0) = Q \tag{22h}$$

$$\dot{\hat{\theta}}^p = \Gamma\, Q^p\, \tilde{\theta}^p,\;\; \tilde{\theta}^p = \theta - \hat{\theta}^p,\;\; \hat{\theta}^p(0) = \hat{\theta} \tag{22i}$$

$$u^p(\tau) \triangleq \kappa(\tau, x_e^p(\tau), \hat{\theta}^p(\tau)) \in \mathbb{U} \tag{22j}$$

$$x_e^p(\tau) \in \mathbb{X}_e,\;\; x_e^p(T) \in \mathbb{X}_{e_f}(\tilde{\theta}^p(T)) \tag{22k}$$

where $\mathbb{X}_e = \left\{ x_e^p \,:\, x^p \in \mathbb{X} \right\}$, $\mathbb{X}_{e_f}$ is the terminal constraint and $\beta \in \{0,\, 1\}$. The effect of the future parameter adaptation is incorporated in the controller design via (22a) and (22k), which results in less conservative worst-case predictions and terminal conditions.

4.2 Implementation Algorithm

Algorithm 1. The finite-time min-max MPC algorithm performs as follows: At sampling instant t_i

(1) **Measure** the current states of the plant $x = x(t_i)$, $\xi = \xi(t_i)$ and obtain the current value of the desired setpoint $x_r = x_r(t_i)$ via the update law (18)

(2) **Obtain** the current value of matrices w, Q and C from

$$\dot{w} = g(x, u) - k_w\, w,\;\; w(t_0) = 0, \tag{23}$$

and

$$\dot{Q} \,=\, w^T w,\;\; Q(t_0) = 0 \tag{24a}$$

$$\dot{C} \,=\, w^T(w\,\theta^0 + x - \hat{x} - \eta),\;\; C(t_0) = 0 \tag{24b}$$

respectively

(3) **If** $det(Q) = 0$ or $cond(Q)$ is not satisfactory update the parameter estimates $\hat{\theta}$ and the uncertainty set $\Theta(t) \triangleq B\left(\hat{\theta}(t), z_\theta(t)\right)$ according to Algorithm 3 in the Appendix.
Else if $det(Q) > 0$ and $cond(Q)$ is satisfactory, set $\beta = 0$ and update

$$\hat{\theta} = Q^{-1}(t_i)C(t_i),\quad z_\theta = 0$$

End

(4) Solve the optimization problem (21,22) and apply the resulting feedback control law to the plant until the next sampling instant

(5) **Increment** $i = i+1$. **If** $z_\theta > 0$, repeat the procedure from step 1 for the next sampling instant. **Otherwise**, repeat only steps 1 and 4 for the next sampling instant.

Since the algorithm is such that the uncertainty set Θ contracts over time, the conservatism introduced by the robustness feature in terms of constraint satisfaction and controller performance reduces over time and when Θ contracts upon θ, the min-max adaptive framework becomes that of a nominal MPC. The drawback of the finite-time identifier is attenuated in this application since the matrix invertibility condition is checked only at sampling instants. The benefit of the identifier, however, is that it allows an earlier and immediate elimination of the robustness feature.

4.3 Lipschitz-based Adaptive MPC

While the min-max approach provides the tightest uncertainty cone around the actual system's trajectory, its application is limited by the enormous computation required to obtain the solution of the min-max MPC algorithm. To address this concern, the robust tracking problem is reposed as the minimization of a nominal objective function subject to "robust constraints".

The model predictive feedback is defined as

$$u = \kappa_{mpc}(t, x_e, \hat{\theta}, z_\theta) = u^*(0) \tag{25a}$$
$$u^*(.) \triangleq \arg\min_{u^p_{[0,T]}} J(t, x_e, \hat{\theta}, z_\theta, u^p, u_r) \tag{25b}$$

where $J(t, x_e, \hat{\theta}, z_\theta, u^p, u_r)$ is given by the optimal control problem:

$$J(t, x_e, \hat{\theta}, z_\theta, u^p, u_r) = \int_0^T L(t, x_e^p, u^p, u_r)d\tau \tag{26a}$$
$$+W(x_e^p(T), z_\theta^p(T)) \tag{26b}$$
$$s.t.\ \forall\tau \in [0,\, T]$$
$$\dot{x}^p = f(x^p, u^p) + g(x^p, u^p)\hat{\theta}, \quad x^p(0) = x \tag{26c}$$
$$\dot{\xi}^p = f(\xi^p, x^p), \quad \xi^p(0) = \xi \tag{26d}$$
$$\dot{x}_r^p = f_r(t, x_r, \hat{\theta}), \quad x_r^p(0) = x_r \tag{26e}$$
$$x_e^p = x^p - x_r \tag{26f}$$
$$\dot{z}_e^p = \beta(\mathcal{L}_f + \mathcal{L}_g\Pi)z_e^p + \|g(x^p, \xi^p, u^p)\|z_\theta, \tag{26g}$$
$$z_x^p(0) = 0 \tag{26h}$$
$$X_e^p(\tau) \triangleq B(x_e^p(\tau), z_e^p(\tau)) \subseteq \mathbb{X}_e, \quad u^p(\tau) \in \mathbb{U} \tag{26i}$$
$$X_e^p(T) \subseteq \mathbb{X}_{e_f}(z_\theta^p(T)) \tag{26j}$$

Since the Lipschitz-based robust controller is implemented in open-loop, there is no setpoint trajectory $x_r(\hat{\theta})$ feedback during the inter-sample implementation. Therefore, the worst-case deviation $z_e^p \geq \max_{\theta\in\Theta}\|x_e - x_e^p\| = \max_{\theta\in\Theta}\|x - x^p\|$. Hence z_e^p given in (26g) follows from

$$\dot{z}_x^p = (\mathcal{L}_f + \mathcal{L}_g\Pi)z_x^p + \|g(x^p, u)\|z_\theta, \quad z_x^p(t_0) = 0 \tag{27}$$

where $\Pi = z_\theta + \|\hat{\theta}\|$. We assume an appropriate knowledge of Lipschitz bounds as follows:

Assumption 6. A set of functions $\mathcal{L}_j : \mathbb{X} \times \mathbb{R}^{n_\xi} \times \mathbb{U} \to \mathbb{R}^+$, $j \in \{f, g\}$ are known which satisfy

$$\mathcal{L}_j(\mathbb{X}, \xi, u) \geq$$
$$\min\Big\{\mathcal{L}_j \;\Big|\; \sup_{x_1,x_2\in\mathbb{X}}\Big(\|j(x_1,\xi,u) - j(x_2,\xi,u)\| - \mathcal{L}_j\|x_1 - x_2\|\Big) \leq 0\Big\},$$

4.4 Implementation Algorithm

Algorithm 2. The finite-time Lipschitz based MPC algorithm performs as follows: At sampling instant t_i

(1) **Measure** the current states of the plant $x = x(t_i)$, $\xi = \xi(t_i)$ and obtain the current value of the desired setpoint $x_r = x_r(t_i)$ via the update law (18)
(2) **Obtain** the current value of matrices w, Q and C from (23) and (24)
(3) **If** $det(Q) = 0$ or $cond(Q)$ is not satisfactory, set $\beta = 1$ and update the parameter estimates $\hat{\theta} = \hat{\theta}(t_i)$ and uncertainty bounds $z_\theta = z_\theta(t_i)$ and $z_\theta^p(T) = z_\theta^p(t_i+T)$ via equation (29)
$$\dot{\hat{\theta}} = \Gamma\,(C - Q\,\hat{\theta}), \quad \hat{\theta}(t_0) = \theta^0, \tag{29}$$
equation (A.1) and equation (30)
$$z_\theta^p(\tau) = \exp^{-\bar{\mathcal{E}}(\tau - t_i)} z_\theta(t_i) \quad \tau \in [t_i,\ t_i + T) \tag{30}$$
where
$$\bar{\mathcal{E}} \geq \mathcal{E}(t_i) = \lambda_{\min}\left(\Gamma Q(t_i)\right).$$
Else if $det(Q) > 0$ and $cond(Q)$ is satisfactory, set $\beta = 0$ and update
$$\hat{\theta} = Q^{-1}(t_i)C(t_i), \quad z_\theta = 0$$
End
(4) **Solve** the optimization problem (25,26) and apply the resulting feedback control law to the plant until the next sampling instant
(5) **Increment** $i = i+1$. **If** $z_\theta > 0$, repeat the procedure from step 1 for the next sampling instant. **Otherwise**, repeat only steps 1 and 4 for the next sampling instant.

Implementing the adaptive MPC control law according to Algorithm 2 ensures that the uncertainty bound z_θ reduces over time and hence, the error margin z_x^p imposed on the predicted state also reduces over time and shrinks to zero when the actual parameter estimate is constructed in finite-time.

4.5 Robust Stability

Robust stability is guaranteed under the standard assumptions that $\mathbb{X}_{e_f} \subseteq \mathbb{X}_e$ is an invariant set, W is a local robust CLF for the resulting time varying system and the decay rate of W is greater than the stage cost L within the terminal set $\mathbb{X}_{e_f}$ in conjunction with the requirement for W to decrease and $\mathbb{X}_f$ to enlarge with decreased parametric uncertainty.

4.6 Enhancing Parameter Convergence

In min-max adaptive formulation, the terminal penalty is parameterized as a function of $\tilde{\theta}$. This ensures that the algorithm will seek to reduce the parameter error in the process of optimizing the cost function and will automatically inject some excitation in the closed-loop system, when necessary, to enhance parameter convergence. However, this is not the case in the Lipschitz-based approach since the control calculation only uses nominal model. To improve the quality of excitation in the closed-loop the proposed excitation cost is

$$J_{\mathcal{E}} = \frac{\beta}{1 + \mathcal{E}_\theta^p(T)} \tag{31}$$

where

$$\mathcal{E}_\theta^p(\tau) = \lambda_{\min}\{Q^p(\tau)\} \quad \text{or} \quad \mathcal{E}_\theta^p(\tau) = \nu^T\, Q^p(\tau)\,\nu \tag{32}$$

with $\nu \in \mathbb{R}^{n_\theta}$ a unit vector. Note that any reduction in the cost function due to $J_{\mathcal{E}}$ implies an improvement in the rank of Q^p. Though, the predicted regressor matrix Q^p differs from the actual matrix Q, a sufficient condition for $Q > 0$ is for $Q^p > z_Q \geq \|Q - Q^p\|$.

5. TWO-LAYER INTEGRATION METHOD

The integration task can also be posed as a two degree of freedom paradigm where the problem is divided into two phases. The first phase deals with generating a state trajectory that optimizes a given objective function while respecting the system's dynamics and constraints, and the second phase deals with the design of a controller that would regulate the system around the trajectory.

The MPC controller design follows that of (21) and (25). The only difference is that rather than solving the setpoint differential equation (18) inside the MPC loop, the measurement of x_r obtained at sampling instants is used as the desired setpoint to be tracked, that is, equations (22e) and (26e) are replaced by

$$\dot{x}_r^p = 0, \;\; x_r^p(0) = x_r. \tag{33}$$

The adaptive controllers are implemented according to Algorithms 1 and 2.

6. MAIN RESULT

The integration result is provided in the following:

Theorem 7. Consider problem (1) subject to system dynamics (2), and satisfying Assumption 1. Let the controller be (21) or (25) with setpoint update law (18) and parameter identifier (34)

$$\hat{\theta}^c(t) = \begin{cases} \hat{\theta}(t), & \text{if } \; t < t_c \\ Q(t_c)^{-1}\, C(t_c), & \text{if } \; t \geq t_c. \end{cases} . \tag{34}$$

If the invertibility condition (equation 35)

$$Q(t_c) = \int_{t_0}^{t_c} w^T(\tau) w(\tau)\, d\tau \succ c_1 I, \tag{35}$$

is satisfied, then for any $\varrho > 0$, there exists constant η_c such that $\lim_{t\to\infty} \|x(t) - x^*(t,\theta)\| \leq \varrho$, with $x^*(t,\theta)$ the unique minimizer of (1). In addition $x \in \mathbb{X}$, $u \in \mathbb{U}$ for all $t \geq 0$.

Proof: We know from from triangle inequality that

$$\|x - x^*(\theta)\| \leq \|x - x_r(\hat{\theta})\| + \|x_r(\hat{\theta}) - x^*_{pa}(\hat{\theta})\| + \|x^*_{pa}(\hat{\theta}) - x^*(\hat{\theta})\| + \|x^*(\hat{\theta}) - x^*(\theta)\| \tag{36}$$

where $x^*_{pa}(\hat{\theta})$ denotes the unique minimizer of the unconstrained problem (15) for $\theta \equiv \hat{\theta}$. Since the MPC controllers guarantees asymptotic convergence of x_e to the origin, we have $\lim_{t\to\infty} \|x - x_r(\hat{\theta})\| = 0$. Also, it follows from Lemma 4, that $\|x_r(\hat{\theta}) - x^*_{pa}(\hat{\theta})\|$ converges exponentially to the origin. Moreover, it is well established that $x^*_{pa}(\hat{\theta})$ converges continuously to $x^*(\hat{\theta})$ as $\eta_c \to \infty$ (Bertsekas, 1995, Proposition 4.1.1). Therefore there exists a class $\mathcal{K}$ function [1] $\alpha_c(\cdot)$ such that

$$\lim_{t\to\infty} \|x^*_{pa}(\hat{\theta}) - x^*(\hat{\theta})\| \leq \alpha_c\left(\frac{1}{\eta_c}\right). \tag{37}$$

The finite-time identification procedure employed ensures that $\hat{\theta} = \theta$ for all $t \geq t_c$, with $t_c < \infty$ and thus $\lim_{t\to\infty} \|x^*(\hat{\theta}) - x^*(\theta)\| = 0$.

Finally, we have

$$\lim_{t\to\infty} \|x(t) - x^*(t,\theta)\| \leq \alpha_c\left(\frac{1}{\eta_c}\right) \tag{38}$$

and the result follows for sufficiently large η_c. The constraint satisfaction claim follows from the feasibility of the adaptive model predictive controllers. ∎

7. SIMULATION EXAMPLE

Consider the parallel isothermal stirred-tank reactor in which reagent A forms product B and waste-product C DeHaan and Guay (2005). The reactors dynamics are given by

$$\begin{aligned}
\frac{dA_i}{dt} &= A^{in}\frac{F_i^{in}}{V_i} - A_i\frac{F_i^{out}}{V_i} - k_{i1}A_i - 2k_{i2}A_i^2, \\
\frac{dB_i}{dt} &= -B_i\frac{F_i^{out}}{V_i} + k_{i1}A_i, \\
\frac{dC_i}{dt} &= -C_i\frac{F_i^{out}}{V_i} + k_{i2}A_i^2,
\end{aligned}$$

where A_i, B_i, C_i denote concentrations in reactor i, k_{ij} are the reaction kinetic constants, which are only nominally known. The inlet flows F_i^{in} are the control inputs, while the outlet flows F_i^{out} are governed by PI controllers which regulate reactor volume to V_i^0.

The economic cost function is the net expense of operating the process at steady state.

$$p(A_i,\, s,\, \theta) = \sum_{i=1}^{2}[(p_{i1}s_i + P_A - P_B)k_{i1}A_iV_i^0 + (p_{i2}s_i + 2P_A)k_{i2}A_i^2V_i^0] \tag{39}$$

where P_A, P_B denote component prices, p_{ij} is the net operating cost of reaction j in reactor i. Disturbances s_1, s_2 reflect changes in the operating cost (utilities, etc) of each reactor. The control objective is to robustly regulate the process to the optimal operating point that optimizes the economic cost (39) while satisfying the following state constraints $0 \leq A_i \leq 3$, $c_v = A_1^2V_1^0 + A_2^2V_2^0 - 15 \leq 0$ and input constraint $0.01 \leq F_i^{in} \leq 0.2$. The reaction kinetics are assumed to satisfy $0.01 \leq k_i \leq 0.2$.

The two-layer approach was used for the simulation. The setpoint value available at sampling instant is passed down to the MPC controller for implementation. The robustness of the adaptive controller is guaranteed via the Lipschitz bound method. The stage cost is selected as a quadratic cost $L(x_e, u_e) = x_e^T\, Q_x\, x_e + u_e^T\, R_u\, u_e$, with $Q_x > 0$ and $R_u \geq 0$.

Terminal Penalty and Terminal Set Design Let $x = [A_1,\ A_2]^T$, $\theta = [k_{11},\, k_{12},\, k_{21},\, k_{22}]^T$ and $u = [F_1^{in},\ F_2^{in}]^T$, the dynamics of the system can be expressed in the form:

$$\dot{x} = -\underbrace{\begin{bmatrix} \dfrac{x_1k_{V1}(\xi_1 - V_1^0 + \xi_3)}{\xi_1} \\ \dfrac{x_2k_{V2}(\xi_2 - V_2^0 + \xi_4)}{\xi_2} \end{bmatrix}}_{f_{p1}} + \underbrace{\begin{bmatrix} \dfrac{A_{in}}{\xi_1} & 0 \\ 0 & \dfrac{A_{in}}{\xi_2} \end{bmatrix}}_{f_{p2}} u - \underbrace{\begin{bmatrix} x_1 & 2x_1^2 & 0 & 0 \\ 0 & 0 & x_2 & 2x_2^2 \end{bmatrix}}_{g} \theta,$$

where ξ_1, ξ_2 are the two tank volumes and ξ_3, ξ_4 are the PI integrators. The system parameters are $V_1^0 = 0.9$, $V_2^0 = 1.5$, $k_{v_1} = k_{v_2} = 1$, $P_A = 5$, $P_B = 26$, $p_{11} = p_{21} = 3$ and $p_{12} = p_{22} = 1$.

A Lyapunov function for the terminal penalty is defined as the input to state stabilizing control Lyapunov function (iss-clf):

$$W(x_e) = \frac{1}{2}\, x_e^T\, x_e \tag{40}$$

[1] A continuous function $\mu : \mathbb{R}^+ \to \mathbb{R}^+$ is of class $\mathcal{K}$ if it is strictly increasing and $\mu(0) = 0$.

Choosing a terminal controller

$$u = k_f(x_e) = -f_{p_2}^{-1}\Big(- f_{p_1} + k_1 x_e + k_2\, g\, g^T\, x_e\Big), \quad (41)$$

with design constants $k_1,\ k_2 > 0$, the time derivative of (40) becomes

$$\dot{W}(x_e) = -k_1\, x_e^T\, x_e - x_e^T\, g\,\theta - k_2 x_e^T\, g\, g^T\, x_e \quad (42)$$

$$\leq -k_1 \|x_e\|^2 + \frac{1}{4k_2}\|\theta\|^2 \quad (43)$$

Since the stability condition requires $\dot{W}(x_e(T)) + L(T) \leq 0$. We choose the weighting matrices of L as $Q = 0.5I$ and $R = 0$. The terminal state region is selected as

$$\mathbb{X}_{e_f} = \{x_e \,:\, W(x_e) \leq \alpha_e\} \quad (44)$$

such that

$$k_f(x_e) \in \mathbb{U}, \quad \dot{W}(T) + L(T) \leq 0, \quad \forall(\theta,\, x_e) \in (\Theta,\, \mathbb{X}_{e_f}) \quad (45)$$

Since the given constraints requires the reaction kinetic θ and concentration x to be positive, it follows that

$$\dot{W} + L = -(k_1 - 0.5)\|x_e\|^2 - x_e^T\, g\,\theta - k_2 x_e^T\, g\, g^T\, x_e \leq 0 \quad (46)$$

for all $k_1 > 0.5$ and $x_e > 0$. Moreover, for $x_e < 0$, the constants k_1 and k_2 can always be selected such that (46) is satisfied $\forall\ \theta \in \Theta$. The task of computing the terminal set is then reduced to finding the largest possible α_e such that for $k_f(.) \in \mathbb{U}$ for all $x \in \mathbb{X}_{e_f}$.

The simulation results are presented in Figures 1 to 3. The phase trajectories displayed in Figure 1 shows that the reactor states obeys the imposed constraints while Figure 2 shows that the actual, unknown setpoint cost $p(t, x_r, \theta)$ converges to the optimal, unknown $p^*(t, x^*, \theta)$. Figure 3 shows the convergence of the parameter estimates to the true values.

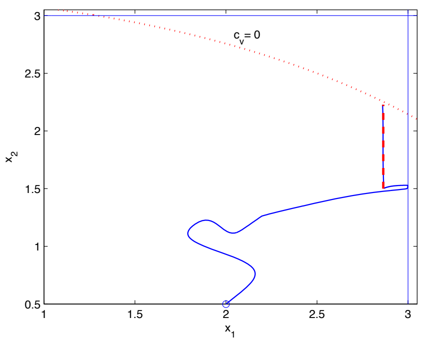

Fig. 1. Phase diagram and feasible state region

8. CONCLUSIONS

This paper provides a formal design technique for integrating RTO and MPC for constrained nonlinear uncertain systems. The solution is based upon the tools and strategies developed in the previous chapters. A single layer and two-layer approaches are presented.

REFERENCES

Dimitri P. Bertsekas. *Nonlinear Programming.* Athena Scientific, Belmont, MA, 1995.

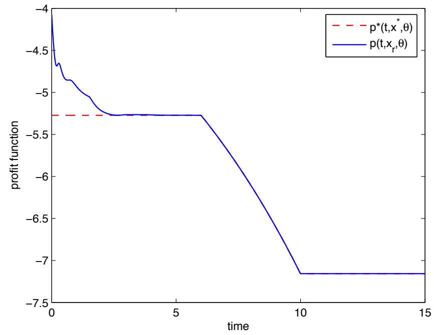

Fig. 2. Optimal and actual profit functions

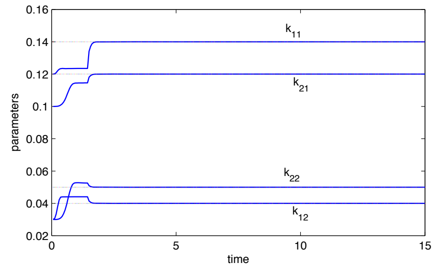

Fig. 3. Unknown parameters and estimates

D. DeHaan and M. Guay. Extremum seeking control of state constrained nonlinear systems. *Automatica*, 41(9): 1567–1574, 2005.

R. Findeisen, H. Chen, and F. Allgöwer. Nonlinear predictive control for setpoint families. In *American Control Conference*, pages 260–264, 2000.

L. Magni. On robust tracking with nonlinear model predictive control. *Int. J. Control*, 75(6):399–407, 2002.

Appendix A. ALGORITHMS

Algorithm 3. Let $\mathcal{E}(\sigma) = \lambda_{\min}(\Gamma Q(\sigma))$, beginning from time $t_{i-1} = t_0$, the parameter and set adaptation is implemented iteratively as follows:

(1) **Initialize** $z_\theta(t_0) = z_\theta^0$, $\Theta(t_0) = B(\hat{\theta}(t_0), z_\theta(t_0))$, $\bar{\mathcal{E}} = \mathcal{E}(t_0) = 0$

(2) **Implement** the following adaptation law over the interval $\tau \in [t_{i-1},\ t_i)$

$$\dot{z}_\theta(\tau) = -\bar{\mathcal{E}} z_\theta(\tau) \quad (A.1)$$

(3) At time t_i, **perform** the updates

$$\bar{\mathcal{E}} = \begin{cases} \mathcal{E}(t_i), & \text{if } \mathcal{E}(t_i) \geq \mathcal{E}(t_{i-1}) \\ \mathcal{E}(t_{i-1}), & \text{otherwise} \end{cases} \quad (A.2)$$

$$\Big(\hat{\theta},\ \Theta\Big) = \begin{cases} \Big(\hat{\theta}(t_i),\ \Theta(t_i)\Big), & \text{if } z_\theta(t_i) - z_\theta(t_{i-1}) \\ & \quad \leq -\|\hat{\theta}(t_i) - \hat{\theta}(t_{i-1})\| \\ \Big(\hat{\theta}(t_{i-1}),\ \Theta(t_{i-1})\Big), & \text{otherwise} \end{cases} \quad (A.3)$$

(4) **Iterate** back to step 2, **incrementing** $i = i + 1$.