ROBUST PREDICTIVE CONTROL BASED ON NEIGHBORING EXTREMALS

S. Gros, B. Srinivasan, and D. Bonvin

Laboratoire d'Automatique
École Polytechnique Fédérale de Lausanne
CH-1015 Lausanne, Switzerland

Abstract: The performance of predictive controller is typically poor when the true plant evolution deviates significantly from that predicted by the model. A robust approach that considers model uncertainty explicitly is then needed. However, it is often difficult to find a single input profile that works for the range of uncertainty considered. Thus, multiple input profiles, i.e. one for each realization of the uncertainty, need to be determined, which is computationally extremely expensive. This paper proposes an alternative approach, based on neighboring extremals, where the multiple input profiles are computed using a simple feedback law, thereby reducing considerably the computational burden. The idea is illustrated via the simulation of a continuous stirred tank reactor and the inverted pendulum on a cart.

Keywords: Predictive control, Robust control, Optimal control, Neighboring extremals.

1. INTRODUCTION

The performance of a predictive controller lies in efficiently tackling problems with constraints and nonlinear dynamics, especially when analytical computation of the control law is difficult (Mayne *et al.*, 2000; Scokaert and Mayne, 1998). Classical predictive control involves recalculating at every sampling instant the input[1] that minimizes a criterion defined over a horizon window in the future, taking into account the current state of the system. Only the first part of the computed optimal input is applied to the system.

A crucial point in predictive control is the extensive use of the dynamic system model that, unfortunately, may not always correspond to the reality. Thus, the predicted state evolution may differ from the actual plant evolution. When the difference between the predicted and the true plant evolutions is significant (which occurs, for example, when the system is unstable), standard predictive control will not be able to provide the desired performance (Mayne *et al.*, 2000). One solution to this problem consists of prestabilizing the system with a simple feedback loop. **Though many methods are available to stabilize nonlinear systems (Khalil, 2002), there are no systematic ways of designing such a feedback (Ronco *et al.*, 2001; Morari and Lee, 1999; Bemporad, 1998)**. Another approach is to cast the problem into a robust framework, where optimization is performed by taking the uncertainty into account explicitly.

Standard robust predictive control computes an input that represent a compromise solution for the range of uncertainty considered (Bemporad

[1] The word input is considered here to be *singular* independent upon whether there are one or several inputs. This choice is made in order to be able later to distinguish between cases with a single or multiple input profiles corresponding to either a single nominal model or several models (one for each realization of the uncertainty).

and Morari, 1999; Lee and Yu, 1997; Kouvaritakis *et al.*, 2000). Furthermore, to prove robust stability, it is important to guarantee that the final state is within some bounded set. When the dispersion of the open-loop predicted state is large, especially in the case of unstable systems, it may not be possible to find a feasible solution to the robust optimization problem. However, due to the feedback introduced by the re-optimizations performed at subsequent sampling instants, the true state dispersion at the end of the prediction horizon will in fact be much smaller than the values given by open-loop prediction. Hence, this feedback needs to be incorporated in the robust predictive control formulation in order to reduce the conservatism and lead to feasible solutions (Warren and Marlin, 2003).

There are two ways of expressing this inherent feedback in a robust optimization framework: (i) use multiple input profiles, i.e. one for each realization of the uncertainy, starting with the next sampling interval (Mayne *et al.*, 2000; Scokaert and Mayne, 1998), and (ii) approximate the inherent feedback by a control law (Bemporad, 1998). The former is computationally expensive, while an ad hoc method has been used for the approximation in the latter.

The idea of using varying state-feedback has already been used in the context of LTV systems using invariant sets (Kouvaritakis *et al.*, 2000). The state-feedback law maintains the state vector inside ellipsoidal feasible regions. This paper proposes a novel way to use state-feedback for MPC. It suggests using the neighboring extremal approach for approximating the feedback inherent to predictive control.

For small deviations from the optimal solution, a linear approximation of the system and a quadratic approximation of the cost are quite reasonable. In such a case, the theory of neighboring extremals (NE) provides a closed-form solution to the optimization problem (Bryson, 1999). Thus, the optimal input can be obtained using state feedback that approximates the feedback provided by explicit numerical optimization.

This approach can also be viewed as a novel way of performing robust optimization with multiple input profiles. Indeed, the proposed scheme optimizes the multiple profiles, one exactly via explicit optimization and the others approximately via the NE-approach. Since only one input profile is optimized explicitly, the computational complexity of the problem reduces considerably, while keeping the advantages of the robust optimization scheme with multiple input profiles. Note that the main emphasis is not in reducing the computational complexity of a general MPC problem as in (Wan and Kothare, 2003), but to really exploit the structure present in the robust optimization problem.

The paper is organized as follows. Background material regarding predictive control and neighboring extremals is presented in Section 2. The classical robust predictive control scheme and robust predictive control with multiple input profiles are presented in Section 3. The robust predictive control based on the NE-approach is presented in Section 4. In Section 5, the various schemes are compared on two illustrative examples. Finally, concluding remarks are given in Section 6.

2. PRELIMINARIES

2.1 Classical Predictive Control

Consider the nonlinear system represented as:

$$\dot{x} = F(x, u, \theta), \qquad x(0) = x_0 \tag{1}$$

where the state x and the input u are vectors of dimension n and m, respectively. x_0 represents the initial conditions, $\theta \in \Theta$ the vector of uncertain parameters, assumed to lie in the admissible region Θ, and F the system dynamics.

In predictive control, the following optimization problem is solved repeatedly at discrete time instants:

$$\min_{u([t_k, t_k+T_f])} J(t_k) = \frac{1}{2} x(t_k+T_f)^T P\, x(t_k+T_f) \tag{2}$$

$$+ \frac{1}{2}\int_{t_k}^{t_k+T_f} \left(x(\tau)^T Q\, x(\tau) + u(\tau)^T R\, u(\tau) \right) d\tau$$

$$\begin{aligned} s.t.\quad & \dot{x} = F(x, u, \theta), \qquad x(t_k) = x_k \\ & x(t_k+T_f) \in \mathcal{X} \end{aligned}$$

where P, Q, and R are positive-definite weighting matrices of appropriate dimensions, $\mathcal{X}$ the bounded region of state space where the final state should be, t_k the present time for which the optimization is performed, T_f the prediction horizon, and x_k the state measured or estimated at the time instant t_k. The optimal input computed by solving (2) is represented by $u^\star([t_k, t_k+T_f])$. The importance of having a terminal cost, and also a bounded region for the final state for the sake of stability, is discused in (Mayne *et al.*, 2000).

Let δ be the sampling period which, in general, is constant. The first part of the optimal input, $u^\star([t_k, t_k+\delta])$, is applied open loop, and the optimization problem is repeated at the time instant t_{k+1}. For implementation purposes, the infinite-dimensional input $u([t_k, t_k+T_f])$ needs to be parameterized using a finite number of decision variables, typically piecewise constant.

2.2 Neighboring Extremals

Including the constraints of the optimization problem (2) in the cost function, the augmented cost function, $\bar{J}$, can be written as (Bryson, 1999):

$$\bar{J} = \Phi(x(t_k + T_f)) + \int_{t_k}^{t_k+T_f} \left(H - \lambda^T \dot{x}\right) dt \quad (3)$$

where $\Phi = \frac{1}{2}x^T P x$, $H = \frac{1}{2}(x^T Q\, x + u^T R\, u) + \lambda^T F(x, u)$, and $\lambda(t) \neq 0$ is the n-dimensional vector of adjoint states (Lagrange multipliers for the system equations), whose dynamics are given by $\dot{\lambda}^T = -H_x$, $\lambda^T(t_k + T_f) = \Phi_x(t_k + T_f)$. The notation $a_b = \frac{\partial a}{\partial b}$ is used. The necessary conditions of optimality read:

$$H_u = u^T R + \lambda^T F_u = 0 \quad (4)$$

At the optimal solution, the first variation of $\bar{J}$ is given by:

$$\Delta\bar{J} = \left(\Phi_x - \lambda^T\right) \Delta x\big|_{t_k+T_f} + \int_{t_k}^{t_k+T_f} \left[\left(H_x + \dot{\lambda}^T\right) \Delta x + H_u \Delta u\right] d\tau \quad (5)$$

where $\Delta x(t) = x(t) - x^\star(t)$ and $\Delta u(t) = u(t) - u^\star(t)$, with x^* and u^* being the optimal state and input, respectively. The conditions of optimality are derived from $\Delta\bar{J} = 0$. The second-order variation of $\bar{J}$ is given by:

$$\Delta^2\bar{J} = \frac{1}{2}\Delta x(t_k + T_f)^T P\, \Delta x(t_k + T_f) + \frac{1}{2}\int_{t_k}^{t_k+T_f} \left[\Delta x^T\ \Delta u^T\right] \begin{bmatrix} H_{xx} & H_{xu} \\ H_{ux} & H_{uu} \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta u \end{bmatrix} d\tau \quad (6)$$

Choosing Δu to minimize $\Delta^2\bar{J}$ under the linear dynamic constraint:

$$\Delta\dot{x} = F_x \Delta x + F_u \Delta u \quad (7)$$

represents a time-varying Linear Quadratic Regulator (LQR) problem, for which a closed-form solution is available:

$$\Delta u(t) = -K(t)\Delta x(t) \quad (8)$$
$$K = H_{uu}^{-1}\left(H_{ux} + F_u^T S\right) \quad (9)$$
$$\dot{S} = -H_{xx} - SF_x - F_x^T S + H_{xu}K + SF_u K$$
$$S(t_k + T_f) = P \quad (10)$$

The details of the development leading to this formulation can be found in (Bryson, 1999). The above controller, termed the neighboring extremal controller, will be used extensively in this paper.

3. EXISTING APPROACHES TO ROBUST PREDICTIVE CONTROL

3.1 Standard Robust Predictive Control

The state x and hence the cost function J are functions of the vector of uncertain parameters θ. For a given value of θ, let x_θ be the state and J_θ the cost function. In robust predictive control, the uncertainty is handled as part of the optimization problem, which is solved repeatedly at discrete time instants:

$$\min_{u([t_k, t_k+T_f])} E_{\theta\in\Theta}[J_\theta(t_k)] \quad (11)$$
$$s.t. \quad \dot{x}_\theta = F(x_\theta, u, \theta) \qquad x_\theta(t_k) = x_k$$
$$x_\theta(t_k + T_f) \in \mathcal{X}, \qquad \forall\theta \in \Theta$$

$$J_\theta(t_k) = \frac{1}{2}x_\theta(t_k + T_f)^T P\, x_\theta(t_k + T_f) + \frac{1}{2}\int_{t_k}^{t_k+T_f} \left(x_\theta(\tau)^T Q\, x_\theta(\tau) + u(\tau)^T R\, u(\tau)\right) d\tau \quad (12)$$

(where $E_{\theta\in\Theta}(x)$ denotes the expectation of x when the random variable θ is in the set Θ). The major difficulty with this formulation is that there may not be a solution, especially when the system is open-loop unstable. The state dispersion might be so large that it is not possible to find a single input $u([t_k, t_k+T_f])$ that satisfies $x_\theta(t_k+T_f) \in \mathcal{X}$, $\forall\theta \in \Theta$.

It should be noted here that a robust control scheme based on the expectation leads to only limited robustness improvement (Nagy and Braatz, 2003). However, this choice of cost function is justified by the fact it is a standard approach in the litterature.

3.2 Robust Predictive Control with Multiple Input Profiles

Though all computations in predictive control are performed open loop, there is inherent feedback due to the state measurement and the re-optimization. This feedback reduces the sensitivity to uncertainty, and thus the state dispersion is often much smaller than what is predicted from an open-loop perspective.

The difficulty with the classical formulation (11) is that it does not take into account the fact that the optimization will be repeated at subsequent time instants. Thus, the idea is to reformulate the robust predictive control problem and include re-optimization in the problem formulation (Mayne *et al.*, 2000):

$$\min_{u([t_k,t_{k+1}]),\bar{u}_\theta([t_{k+1},t_k+T_f])} E_{\theta\in\Theta}[J_\theta(t_k)] \quad (13)$$
$$s.t. \quad \dot{x}_\theta = F(x_\theta, u_\theta, \theta) \qquad x_\theta(t_k) = x_k$$
$$x_\theta(t_k + T_f) \in \mathcal{X}, \qquad \forall\theta \in \Theta$$

$$u_\theta = \begin{cases} u & \text{if } t_k \le t < t_{k+1} \\ \bar{u}_\theta & \text{if } t_{k+1} \le t \le t_k + T_f \end{cases} \quad (14)$$

$$J_\theta(t_k) = \frac{1}{2}x_\theta(t_k+T_f)^T P\, x_\theta(t_k+T_f) \quad (15)$$
$$+ \frac{1}{2}\int_{t_k}^{t_k+T_f} \left(x_\theta(\tau)^T Q\, x_\theta(\tau) + u_\theta(\tau)^T R\, u_\theta(\tau)\right) d\tau$$

where $\bar{u}_\theta$ is the input for the realization θ during the prescribed interval. This means that the manipulated variables consist of one set of inputs for all realizations of θ between t_k and t_{k+1}. However, between t_{k+1} and $t_k + T_f$, different sets of inputs are required for different realizations of θ. This problem is computationally expensive to solve since $\bar{u}_\theta([t_{k+1}, t_k + T_f])$ needs to be optimized for every realization of θ. Note that, though $\bar{u}_\theta([t_{k+1}, t_k + T_f])$ is important for calculating $u([t_k, t_{k+1}])$, it will never be implemented on the real system. What will be implemented is the part $u([t_k, t_{k+1}])$, which represents the compromise input for all the uncertainty realizations considering all future possibilities.

4. ROBUST PREDICTIVE CONTROL BASED ON NEIGHBORING EXTREMALS

In order to avoid having to compute the optimal input for many different realizations of θ, a relationship between the uncertain parameters θ and the optimal input is needed. The idea proposed in this paper is to use the NE-approach. As seen in Section 2, the NE-approach provides the following relationship:

$$u_\theta([t_{k+1}, t_k + T_f]) = u_{\theta_0}([t_{k+1}, t_k + T_f])$$
$$-K(t)\Delta x([t_{k+1}, t_k + T_f]) \quad (16)$$

where θ_0 is the nominal parameter vector. The optimization problem then becomes:

$$\min_{u([t_k,t_k+T_f])} E_{\theta\in\Theta}[J_\theta(t_k)] \quad (17)$$
$$s.t. \quad \dot{x}_{\theta_0} = F(x_{\theta_0}, u, \theta_0), \quad x_{\theta_0}(t_k) = x_k$$
$$\dot{x}_\theta = F(x_\theta, u_\theta, \theta), \quad x_\theta(t_k) = x_k, \qquad \forall\theta \ne \theta_0$$
$$x_\theta(t_k + T_f) \in \mathcal{X}, \qquad \forall\theta \in \Theta$$

$$u_\theta = \begin{cases} u & \text{if } t_k \le t < t_{k+1} \\ u - K(x_\theta - x_{\theta_0}) & \text{if } t_{k+1} \le t \le t_k + T_f \end{cases}$$

$$J_\theta(t_k) = \frac{1}{2}x_\theta(t_k+T_f)^T P\, x_\theta(t_k+T_f)$$
$$+ \frac{1}{2}\int_{t_k}^{t_k+T_f} \left(x_\theta(\tau)^T Q\, x_\theta(\tau) + u_\theta(\tau)^T R\, u_\theta(\tau)\right) d\tau$$

Note that the decision variables in (17) are only $u([t_k, t_k + T_f])$ for the nominal plant and not $u_\theta([t_k, t_k + T_f])$ for all realizations. The input profiles for realizations other than the nominal one are computed using the NE-controller. The parameterization of the nominal input can be chosen conveniently.

The NE-approach can be interpreted from two viewpoints: (i) From the view point of feedback, it is an approximation of the inherent feedback provided by the predictive control itself; (ii) From the point of view of robust predictive control with multiple input profiles, and given the nominal optimal input profile, the NE-approach computes to a first-order approximation the optimal input profiles that correspond to the various realization of θ. Note that, as in Subsection 3.2, the input computed via the NE-approach will never be implemented on the true system since it is computed for the time interval $[t_{k+1}; t_k + T_f]$. In that sense, the NE-feedback is a fictitious one that serves only a computational purpose.

The proposed NE-approach is only an approximation of robust control with multiple input profiles and thus should have an inferior performance. However, in the examples that have been worked out, the proposed NE-approach often led to a slight improvement in performance over robust control with multiple input profiles. This can be attributed to the fact that robust control with multiple input profiles requires the solution of an optimization problem with a large number of decision variables and, thus, often gets stuck in a local minimum due to sensitivity issues. Thus it might be advantageous to use a well-posed feedback law instead of a poorly posed open-loop problem.

Though this paper presents no stability proof for the proposed robust predictive controller based on the NE-approach, many pointers indicate that robust stability can indeed be established. The steps to be followed are: (i) Stability of the robust predictive control with multiple input profiles (Mayne *et al.*, 2000), (ii) Proof that robust predictive control based on NE is a first-order approximation of that with multiple input profiles (see preliminary work in this direction by (Ronco *et al.*, 2001)), and (iii) Effect of the approximation error on stability.

The computation of the feedback law $K(t)$ adds some load to the optimization algorithm. It requires the computation of the system adjoints $\lambda(t)$ and the sweep matrix $S(t)$. These computations are performed using backward integration based on the optimal input found for the nominal system. Integration of the adjoints can require some precaution as these variables can be unstable. However, experience has shown

that a appropriate choice of the cost function facilitates the integration significantly. Computing the optimal input for the nominal system and integrating the adjoints and sweep matrix is considerably less demanding than computing the optimal input for several models in the model set.

5. ILLUSTRATIVE EXAMPLES

In this section two examples will be presented. The first one is a chemical engineering example, a continuous stirred tank reactor, where the standard predictive control stabilizes the system but with poor performance. The standard robust predictive control in fact unstable, and the proposed methodology gives good performance. The second one is a mechanical engineering example, an inverted pendulum on the cart, where the proposed method or robust control with multiple input profiles is essential even for stability. Also, since the second example is simpler, dispersion plots are included to illustrate the effect of the uncertainty and its influence on optimization.

5.1 Control of a Continuous Stirred Tank Reactor

5.1.1. System Model: Control of the concentration of a species in a CSTR with constant cooling jacket temperature is considered. The manipulated variable is the inlet feed rate. The model equations are:

$$\begin{aligned} \dot{c}_A &= -k_0 c_A e^{\frac{-E}{RT}} + \frac{F}{V}(c_{A_{in}} - c_A) \qquad (18) \\ \dot{T} &= \frac{(-\Delta H)}{\rho c_p} k_0 c_A e^{\frac{-E}{RT}} + \\ &\quad \frac{F}{V}(T_{in} - T) + \frac{UA}{V\rho c_p}(T_c - T) \end{aligned}$$

where c_A is the concentration of species A, T the reactor temperature, T_{in} the inlet temperature, T_c the cooling jacket temperature, F the feed rate of A, k_0 the pre-exponential factor, E the activation energy, R the gas constant, V the volume, ΔH the reaction enthalpy, ρ the density, c_p the heat capacity, U the heat transfer coefficient, and A the heat transfer area. The parameter values are listed in Table 1 and are taken from (Eker and Nikolaou, AIChE J. 48(9), 1957-80, 2002).

The uncertainty considered is on the pre-exponential factor k_0 that can take any value within the set $[5.66 \times 10^7, 8.5 \times 10^7]$ with a uniform probability distribution. The actual value of k_0 for the simulated reality is $k_0 = 8.14 \times 10^7 [1/h]$. Note that the simulated reality is more reactive than the nominal model ($k_0^{nom} = 7.08 \times 10^7\,[1/h]$).

The goal of the predictive control scheme is to drive the system from the equilibrium point $c_A(0) = 1586.08\,[mol/m^3]$, $T(0) = 544.72\,[K]$ to the equilibrium point $c_A^{ref} = 4786.23\,[mol/m^3]$, $T^{ref} = 487.79\,[K]$.

V	1.36	m^3
$c_{A,in}$	8008	mol/m^3
k_0	7.08×10^7	1/h
E/R	8375	K
T_{in}	373.3	K
T_c	532.6	K
ρ	800.8	kg/m^3
c_p	3140	J/(kgK)
UA	7.04×10^6	J/(hK)
$(-\Delta H)$	69775	J/mol

Table 1. Model parameters

5.1.2. Integral Control: As the parameter k_0 varies, the equilibrium point of the system shifts, i.e. when $F = F^{ref}$ is applied it no longer gives $c_A = c_A^{ref}$ and $T = T_{ref}$. Therefore, the predictive control schemes seek a compromise between meeting $F = F^{ref}$ at the end and $c_A = c_A^{ref}$. This causes a steady state error that can be eliminated by using integral control. For this a supplementary state I is included which has the following dynamics:

$$\dot{I} = c_A^{ref} - c_A \qquad I(0) = 0 \qquad (19)$$

If a non-zero weighting of I is included in the cost function, if the predictive control scheme converges, it has to converge to c_A^{ref}. This can be shown by contradiction, since $c_A \neq c_A^{ref}$ will cause I and the cost function to grow indefinitely. **An important disadvantage of integral control is the problem of wind up. It can occur e.g. if the input saturates while the output has not reached the setpoint. In our example though, for all models of the model set, the setpoint does clearly not require inputs that are outside the valid range. Therefore an integral control is justified as it is the simplest method to solve the problem of permanent error.**

5.1.3. Control Parameters: All schemes share the following features: (i) re-optimization of the input at each sampling instant δ, for all techniques $\delta = 0.2\,[h]$, (ii) prediction horizon, $T_f = 1\,[h]$, (iii) control horizon = sampling period, i.e. the control sequence for $t_k \le t \le t_k + \delta$ is applied, leaving the rest of the sequence unused, (iv) the input parameterization considers $F(t)$ being piecewise constant over time intervals of width δ, and (iv) $P = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 4 \end{bmatrix}$, $Q = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$ and $R = 10^3$.

Note that the temperature is not pushed to its reference, as only the concentration control is considered here. The expectation $E_{\theta \in \Theta}(J_\theta)$ is approximated as $\sum_{k=1}^{3} \frac{1}{3} J_{k_0^k}$, with $k_0^1 = 5.66 \times 10^7\,[1/h]$, $k_0^2 = 7.08 \times 10^7\,[1/h]$ and $k_0^3 = 8.5 \times 10^7\,[1/h]$.

5.1.4. Classical Predictive Control: The classical non-robust re-optimization scheme (2) is applied first. The control is computed from a single nominal model with $k_0^{nom} = 7.08 \times 10^7\,[1/h]$ and applied to the simulated reality with $k_0 = 8.14 \times 10^7\,[1/h]$. The simulation results are displayed in Figure 1, which shows that this scheme ultimately converges, but applies unnecessary bang-bang control on the system. This in turn causes large variations in c_A which in fact goes to zero for a short time (consumption of all A due to no feed and high temperature). An improvement in performance is sought by resorting to robust predictive control methods.

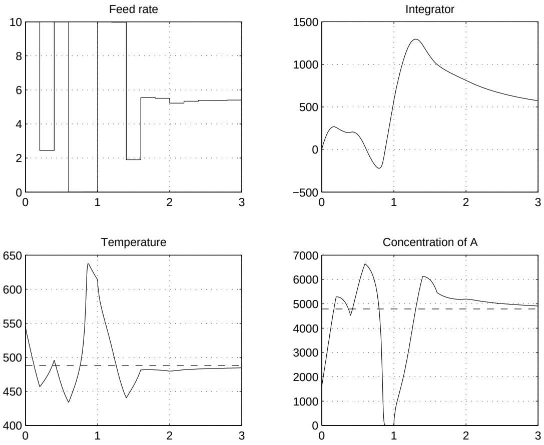

Fig. 1. Classical Predictive Control

5.1.5. Standard Robust Predictive Control: The robust predictive control scheme considered here uses three models ($k_0^1 = 5.66 \times 10^7\,[1/h]$, $k_0^2 = 7.08 \times 10^7\,[1/h]$, $k_0^3 = 8.5 \times 10^7\,[1/h]$). The optimization scheme (11) computes a single input profile for all three models. The simulation results are displayed in Figure 2. They show that this approach does not even converge. The optimization is unable to find a single input profile that works well with all three models. Hence, the predicted value of the cost function is high and does not decrease with the number of re-optimizations. No final constraint is imposed since this optimization is unable to provide a feasible solution.

5.1.6. Robust Control with Multiple Input Profiles: The robust control scheme with multiple input profiles (13) is studied next. The same three models are used but, at each re-optimization, three different control sequences are computed, i.e. one for each model. The input for the first sampling interval is the same for all three models, but distinct for each model thereafter. The results of the simulation are displayed in Figure 3. The computational burden is very heavy, the average duration of one re-optimization is approximately 1.21 $[h]$. Also, since the number of decision variables and the number of simulations to be performed increase with the number of realizations considered, the computational time increases *quadratically* with the number of realizations.

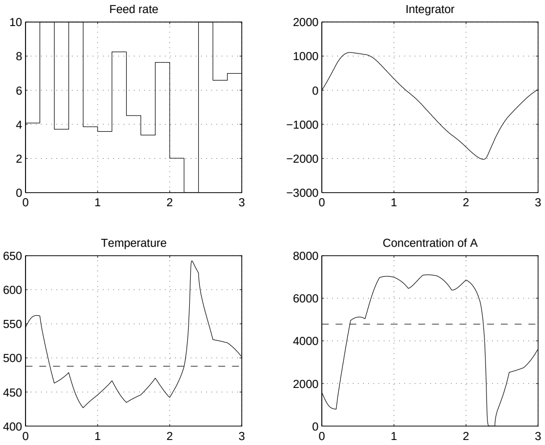

Fig. 2. Standard Robust Predictive Control

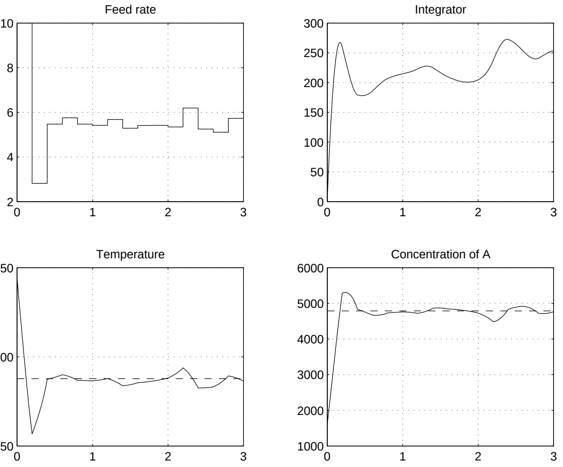

Fig. 3. Robust Control with Multiple Input Profiles

It can be noted that the convergence is not perfect since there remain oscillations in the input and states. This is due to the numerical difficulties and the sensitivity issues associated with this scheme. Indeed, since the optimization has a large number of variables resulting in a lack of sensitivity of the cost function, the numerical optimization algorithm struggles to find the optimum and stops at sub-optimal values.

5.1.7. Robust Control based on Neighboring Extremals: The proposed approach (17) is used, where the input consists of two parts. The first part, for $t = [t_k, t_k + \delta]$, is a piecewise constant input that is common to the three models. The second part, for $t = [t_k+\delta, t_k+T_f]$, is generated by a NE-controller and differs for each model. At each re-optimization, an optimal input is computed for the nominal model ($k_0 = 7.08 \times 10^7\,[1/h]$), using a piecewise constant parametrization. Based on this input, a NE-controller is designed to generate a trajectory for each of the three models in the interval $t = [t_k + \delta, t_k + T_f]$.

The simulation results, displayed in Figure 4, show that this approach works well on this system. The computational burden is much lower than with multiple input profiles: the average duration of a re-optimization is of the order of $0.1\,[h]$, a considerable reduction compared to scheme (13) (factor 12), making the real-time implementation realistic (reoptimization time $< \delta$). Moreover, with the NE-approach, the computational time increases *linearly* with the number of models used to represent the uncertainty.

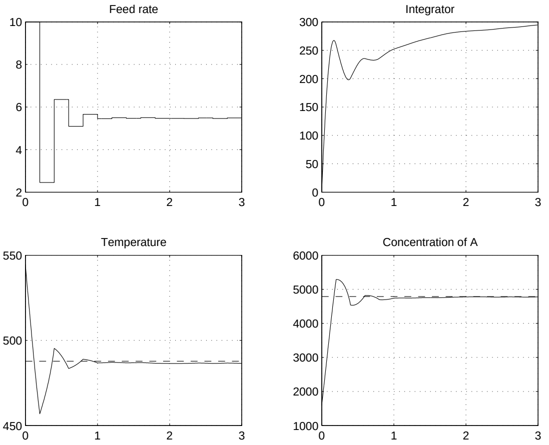

Fig. 4. Robust Control with Neighboring Extremals

5.2 Control of an Inverted Pendulum

5.2.1. System Model: This section illustrates the application of both classical and robust predictive control schemes to an inverted pendulum on a cart, ignoring the cart dynamics (Ronco *et al.*, 2001). The model equations are:

$$\dot{x}_1 = x_2 \tag{20}$$

$$\dot{x}_2 = \frac{m\,l}{J}[sin(x_1)\,g - cos(x_1)u] \tag{21}$$

where x_1 is the pendulum angle, x_2 its rotational velocity, and u the control torque. The control objective consists of regulating the pendulum around the upright position, starting from the downward position $x_0 = [\pi \quad 0]$. The following numerical values are used: $m = 1\,[kg]$, $g = 9.81\,\left[\frac{m}{s^2}\right]$, $l = 1\,[m]$ and $J = 1\,[kg \cdot m^2]$. In addition, the control is constrained, $5\,\left[\frac{m}{s^2}\right] < u < 5\,\left[\frac{m}{s^2}\right]$. For all techniques, the same sampling period of $\delta = 0.2\,[s]$ is used.

In the following simulations, it is assumed that the mass of the pendulum is unknown but lies somewhere between $0.5\,[kg]$ and $1.5\,[kg]$, with a uniform probability distribution. The expectation $E_{\theta\in\Theta}\,(J_\theta)$ is approximated as $\sum_{k=1}^{3}\frac{1}{3}J_{m_k}$, with $m_1 = 0.5\,[kg]$, $m_2 = 1\,[kg]$ and $m_3 = 1.5\,[kg]$. The predictive control schemes are applied to a simulated reality with a mass $m = 1.32\,[kg]$, thus making the real system slower than the nominal model ($m = 1\,[kg]$).

5.2.2. Control Parameters: All schemes share the following features: (i) re-optimization of the input at each sampling instant, (ii) prediction horizon, $T_f = 1s$, (iii) control horizon = sampling period, i.e. the control sequence for $t_k \le t \le t_k+\delta$ is applied, leaving the rest of the sequence unused, (iv) $x(t_k+T_f) \in \mathcal{X}$ was not enforced, (v) $P = 10\,I$, $Q = 10^{-2}$, $R = 0.02$, where I is the identity matrix, and (vi) the input parameterization considers $u(t)$ constant over the time interval $[t_k, t_k + \delta]$. The rest of the input trajectory is obtained using the shooting method (Lewis, 1986), with the initial conditions of the adjoint variables serving as parameters.

5.2.3. Simulation results: The classical non-robust re-optimization scheme (2) is applied first (Figure 5, left column). It can be seen that the approach does not work since the sampling period is too large for the control scheme to converge.

The robust predictive control scheme (11) is considered which computes a single input profile for all three models mentioned above (Figure 5, right column). This approach does not work either since it is unable to find a single input profile that works well with all three models.

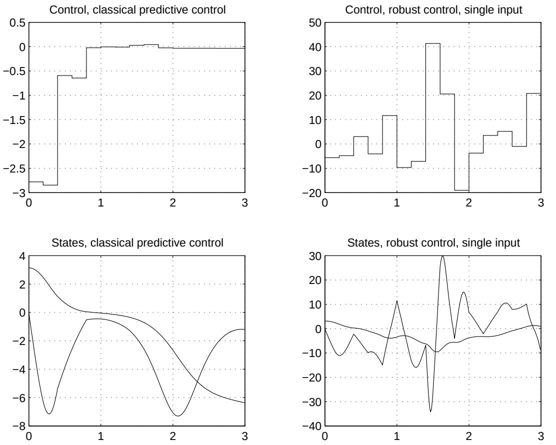

Fig. 5. Standard Robust Predictive Control (left column) and Robust predictive control (right column)

The robust control scheme with multiple input profiles (13) is studied next (Figure 6 left column).

The computational burden is very heavy, the duration of one re-optimization (when the system is not close to the reference) is approximately 5200 seconds. The convergence is quite good.

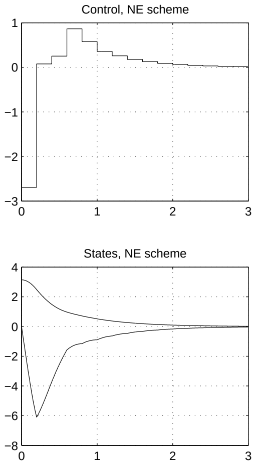

Fig. 6. Robust Control with Multiple Input Profiles (left column) and Robust Control with Neighboring Extremals (right column)

The proposed approach (17) is used. At each re-optimization, an optimal input is computed for the nominal model ($m = 1\,[kg]$), using a shooting method. Based on this optimal input, a NE-controller is designed to generate the optimal trajectory for each of the three models in the interval $t = [t_k + \delta, t_k + T_f]$. The simulation results, displayed in Figure 6 right column, show that this approach works well on this system. The computational burden is much lower than with multiple input profiles: the duration of a re-optimization (when the system is not close to the reference point) is of the order of 700 seconds (reduction by a factor 7.5).

5.2.4. Dispersion plots In figure (7) the strong deviation taken by the system from the trajectory computed according to the model appears clearly. In figure (8) is shown the dispersion for the different models when one single input is computed that minimize the expectation of the costs for the three models. Though all three models get in a neighborhood of $(0, 0)$, a dispersion clearly appears, showing that a single input is not able to drive the three systems to the reference point. In figure (9) is shown the effect of a multiple input profile scheme computed by the NE controller. While the first part of the input is unique, the rest of the sequence is made of three different inputs, one for each system. These inputs are generated by the NE-controller. It clearly appears that the NE-controller is able to drive the three systems to the point $(0, 0)$. Figure (10) shows the effect of the multiple input profile scheme, that also drives the three models to the point $(0, 0)$.

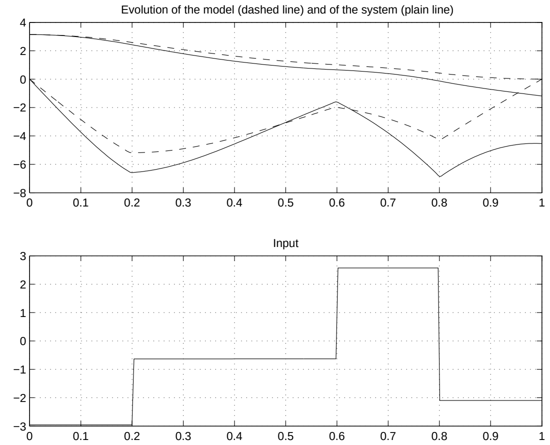

Fig. 7. Dispersion between the model and the system

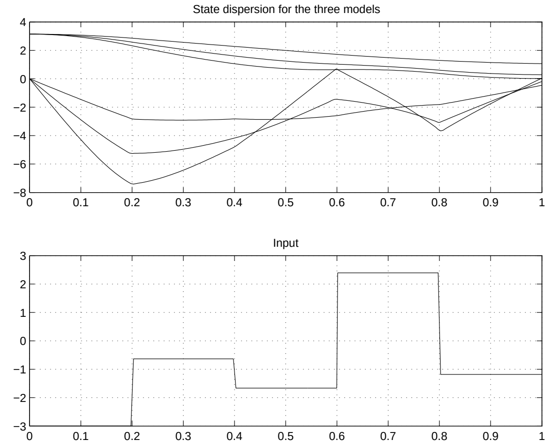

Fig. 8. Dispersion of the various models when 1 input is applied

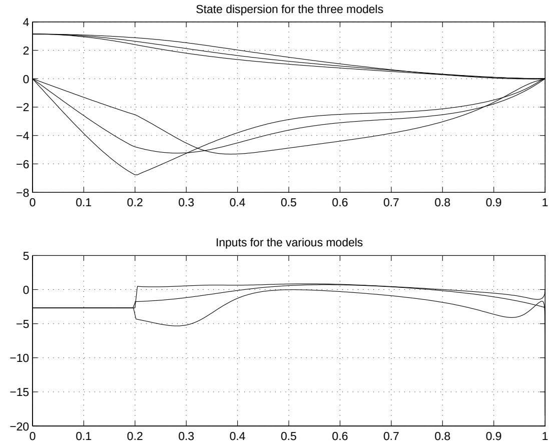

Fig. 9. Dispersion of the various model with a multiple input generated by the NE-controller

6. CONCLUSION

This paper has addressed the problem of robust predictive control of systems for which the open-loop prediction of the future state evolution leads to very conservative results. The fact that there will be re-optimizations needs to be incorporated

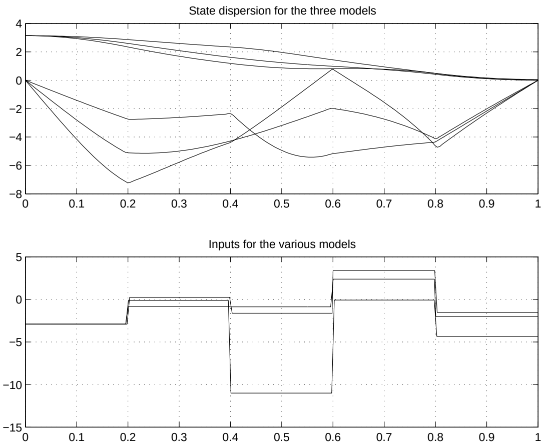

Fig. 10. Dispersion of the three models with the piecewise multiple input

in the predictions, and this was done in this paper using the NE-approach. The proposed approach was illustrated and compared to other approaches on a simple unstable mechanical example.

The stability and performance of the NE-based robust predictive control scheme have not been addressed in this paper. These issues will form the subject of future research. The main issues involved therein will be how good the approximation is and how the approximation error influences stability. In this paper, the feedback computed using the NE-approach was only used as a fictitious one for computational purposes. Its use for implementation is another promising research direction.

REFERENCES

Bemporad, A. (1998). Reducing conservatism in predictive control of constrained systems with disturbances. In: *37th IEEE Control and Decision Conference*. Tampa, FL. pp. 1384–89.

Bemporad, A. and M. Morari (1999). *Robust Model Predictive Control: a Survey*. Springer Verlag.

Bryson, A. E. (1999). *Dynamic Optimization*. Addison-Wesley, Menlo Park, California.

Khalil, H. K. (2002). *Nonlinear Systems*. Prentice-Hall.

Kouvaritakis, B., J. A. Rossiter and J. Schuurmans (2000). Efficient robust predictive control. *IEEE Trans. Automat. Contr.* **45**(8), 1545–49.

Lee, J. H. and Z. Yu (1997). Worst-case formulations of model-predictive control for systems with bounded parameters. *Automatica* **33**(5), 763–781.

Lewis, F. L. (1986). *Optimal Control*. John Wiley, New York.

Mayne, D. Q., J. B. Rawlings, C. V Rao and P. O. M. Scokaert (2000). Constrained model predictive control: Stability and optimality. *Automatica* **36**(6), 789–814.

Morari, M. and J. H. Lee (1999). Model predictive control: Past, present, and future. *Comp. Chem. Eng.* **23**, 667–682.

Nagy, Z. K. and R. D. Braatz (2003). Robust nonlinear model predictive control of batch processes. *AIChE Journal* **49**(7), 1776–1786.

Ronco, E., B. Srinivasan, J. Y. Favez and D. Bonvin (2001). Predictive control with added feedback for fast nonlinear systems. In: *European Control Conference*. Porto, Portugal. pp. 3167–3172.

Scokaert, P. O. and D. Q. Mayne (1998). Minmax feedback model predictive control for constrained linear systems. *IEEE Trans. Automat. Contr.* **43**, 1136–1142.

Wan, Z. and M. V. Kothare (2003). An efficient off-line formulation of robust model predictive control using linear matrix inequalities. *Automatica* **39**, 837–846.

Warren, A. L. and T. E. Marlin (2003). Improved output constraint-handling for mpc with disturbance uncertainty. *Proceedings of the American Control Conference* **6**, 4573–4578.