An MPC Approach to Dual Control

Tor Aksel N. Heirung * B. Erik Ydstie ** Bjarne Foss *

* Department of Engineering Cybernetics, Norwegian University of Science and Technology, N-7491 Trondheim, Norway (e-mail: heirung@stik.ntnu.no).
** Department of Chemical Engineering, Carnegie Mellon University, Pittsburgh, PA 15213, USA.

Abstract: We present a model predictive control (MPC) approach to solve the dual adaptive control problem. The cost function minimized by the controller rewards probing the system for information when the parameter estimates are poor. The control algorithm is designed to handle poorly identified models and excites the system so that information can be gathered to achieve the optimal trade-off between process control and identification. This excitation is achieved without requiring the input to be persistently exciting; rather, the probing objective is based on an exact formulation of the expected value of the output error at the first time stage. The resulting expression is also used for the second time stage; this ensures that a proper trade-off between excitation and output regulation is maintained. The algorithm can be viewed as the merging of adaptive control with MPC and its design can easily be implemented with modifications to an existing MPC. As an example we consider a first-order linear process system with two unknown parameters. Our proposed algorithm probes the system even when the output error is small and quickly gathers enough information to correctly identify the unknown plant parameters.

Keywords: MPC, dual control, adaptive control, optimal control, nonlinear programming, parameter estimation, system identification, stochastic systems.

1. INTRODUCTION

Maintaining a good model of a controlled plant is an important challenge in the process industries, model quality being one determining factor for the performance of a model predictive controller (MPC). Performing experiments to generate data suitable for system identification is not always practical due to factors such as time constraints, the expertise needed, and expensive operational disruption. Model parameters are therefore commonly estimated using data collected during normal operation; however, recorded process data may be insufficiently informative for system identification or it can be difficult to locate the informative portions of a large data set.

Feldbaum (1961) was the first to recognize that an optimal controller for a system with unknown parameters has two conflicting tasks: directing the output toward a reference, and exciting the system for learning purposes so that better control decisions can be made in the future. In its simplest form the dual control problem can be seen as finding the sequence of controls \( \{u(t) : t = 0, 1, 2, \ldots\} \) that minimizes the control objective

\[
J_{\infty} = E \left\{ \sum_{t=0}^{\infty} y(t)^2 + \mu u(t)^2 \right\}
\]

given data collected up until \( t = 0 \), where \( \mu \geq 0 \) is a weighting parameter. This formulation does not include the notion of a model of the system. Thus we expect that the controller should be capable of exploring the system to find out the causal relationship between input \( u \) and output \( y \). The dynamic properties of the system emerge as exploration takes place. This knowledge can be stored, for example in the terms of tables or as parameters in a linear or nonlinear model. We would expect the performance of the control system to improve as more data is gathered. A major contribution of the field of adaptive control was to show that the seemingly impossible problem of even stabilizing an unknown system could be solved for some classes of systems (Märtensson, 1985). However, claims of optimality were not made. One important property of a dual controller is that it is optimal in the sense that it finds the best trade-off or balance between control and excitation.

Dual control laws have been computed for very simple systems using dynamic programming (Åström and Helmersson, 1986), but the optimization problem quickly becomes intractable as the number of unknown parameters increases. Lee and Lee (2009) approach the problem of exponential growth in computational requirements using approximate dynamic programming; Bayard and Schmitzky (2010) developed a sampling-based approach to the dual control problem based on forward dynamic programming and particle filtering. Other approximations have been proposed to solve the problem of combined estimation and control. Some of these are motivated by adaptive control and they lead to the idea of generating exiting signals (Radenkovic and Ydstie, 1995). These approaches led to the idea of iterative adaptive control where instabilities
observed in adaptive controllers were controlled by supplying sufficient excitation (Gevers, 2002). Around the same time the problem was approached from the point of view of Model Predictive Control and an approach termed Model Predictive Control and Identification (MPCI) was introduced by Geneci and Nikolaou (1996) and later extended in several publications by Nikolaou and coworkers, notably Shouche et al. (2002). The MPCI approach is based on parameterizing the input as a sum of sinusoids with prescribed frequencies and finding the optimal amplitudes; this leads to an input that is persistently exciting. While persistent excitation guarantees that parameter estimates converge exponentially when a recursive weighted least-squares algorithm is used (Johnstone et al., 1982), the excitation may be excessive. Marafioti et al. (2012) developed a persistently exciting MPC using techniques similar to those used by Geneci and Nikolaou (1996).

The papers above show that the adaptive control and MPC control approaches appear to be converging. However, the previous approaches that have been developed, whether from an MPC or an adaptive control perspective, have not taken advantage of the fact that these perspectives can be blended by mixing MPC and learning directly into the MPC objective as originally proposed by Feldbaum when he developed the idea of dual control. While the original formulation of dual control may be intractable due to strong nonlinearity and the postulation of an infinite horizon, it seems reasonable to believe that good approximations can be obtained by considering finite horizon formulations. This approach has been successfully applied in the areas of nonlinear MPC and Moving Horizon Estimation in the form of finite horizon approximations to the infinite horizon problem, formulated within the context of dynamic programming. By combining the objectives into one, as suggested by the dual control formulation, we develop an approach that does not require external excitation to excite the system unnecessarily since excitation is explicit in the sense that excitation is part of the control objective. In this case the objective above is restated so that

\[ J_N(t) = E \left\{ \sum_{k=0}^{N} (y(t + k)^2 + \mu u(t + k)^2 \right\} + J(t + N) \bigg| \mathcal{Y}(t) \right\} \]  

where \( E \{ \cdot | \mathcal{Y}(t) \} \) denotes the conditional expectation given all data gathered up to time \( t \), \( \mathcal{Y}(t) \); \( N \geq 1 \) is the prediction horizon; and \( J(t + N) \) is the cost to go that may not include the exploration component in order to make the problem computationally tractable. This control can then be computed using nonlinear programming and it will be implemented using the idea of receding horizon. Setting \( N = 1 \) give the classical adaptive control approach since and no exploration is provided. Extending the prediction horizon to 2 and beyond provide control signals that include a trade-off between exploration and control.

In this paper we develop a case study to illustrate the application of Dual Model Predictive Control (DMPC) and we give particular attention to systems and formulations that cause certainty-equivalence based adaptive controllers to fail. The most common problem is an input gain estimated to be zero; other problems include models of dynamic systems where the dynamic component is missing and pole-zero cancellations in transfer functions.

This article is organized as follows: The control problem is described in Section 2, followed by a discussion of the control algorithm in Section 3. Sections 4 and 5 discuss implementation and provides an example, respectively. We conclude and discuss future work in Section 6.

2. PROBLEM FORMULATION

We consider control of stably invertible, single input, single output, linear, time-invariant systems in discrete time with unknown parameters, disturbed by a sequence of independent zero-mean Gaussian variables with variance \( r \). Systems of this type can be formulated as autoregressive process with exogenous input (ARX):

\[ y(t) + a_1 y(t - 1) + \cdots + a_n y(t - n_a) = b_0 u(t - 1) + \cdots + b_{n_b - 1} u(t - n_b) + v(t) \]  

where \( t \) is a discrete time instant (integer), \( y(t), u(t), \) and \( v(t) \) are the system output, input, and disturbance respectively at time \( t \), and \( a_1, \ldots, a_n \) and \( b_1, \ldots, b_{n_b} \) are the unknown system parameters with \( b_0 \neq 0 \). The system is stably invertible so we can set \( \mu = 0 \) in the control objective and focus on controllers of the type

\[ u(t) = \frac{1}{b_0} (-a_1 y(t) - \cdots - a_n y(t - n_a) - b_1 u(t - 1) - \cdots - b_{n_b - 1} u(t - n_b + 1) \]  

This controller gives the optimal control when the parameters are precisely known. When they are not known we need to consider algorithms that estimate the parameters and implement controllers that in some way trade-off the learning and control.

In adaptive control we use the past information to estimate the parameters, for instance by solving a least squares problem. Subject to the assumptions made above we can guarantee that these estimates are optimal in the sense that they give the Best Linear Unbiased Estimate (BLUE) (Ljung, 1999). We can then use the estimated parameters to calculate the control law in Equation (4) with the knowledge that the estimates we use are the best estimates we can obtain using current information. This is the essence of the certainty-equivalence approach to adaptive control and it has been shown that this in fact can give optimal controls asymptotically. The approach however ignores the possibility of generating exploration signals that can speed up convergence by manipulating control signals to gain better knowledge about the parameters.

The ARX system (3) can be written in the compact form

\[ y(t) = \varphi^T (t - 1) \theta + v(t) \]  

where

\[ \theta = [a_1, \ldots, a_n, b_1, \ldots, b_{n_b}]^T \]

is a vector containing all system parameters, and

\[ \varphi(t - 1) = [-y(t - 1), \ldots, -y(t - n_a) \]  

\[ u(t - 1), \ldots, u(t - n_b) \]  

is a regression vector containing past inputs and outputs. We can also write (3) as

\[ A(q^{-1}) y(t) = B(q^{-1}) u(t) \]
where $A$ and $B$ are polynomials in the backwards shift operator $q^{-1}$. It is necessary for controllability and observability that $A(q^{-1})$ and $B(q^{-1})$ are coprime.

We let $\mathcal{Y}(t)$ denote all inputs and outputs recorded up until the present time $t$. That is,

$$\mathcal{Y}(t) = \{ u(t), u(t-1), \ldots, y(t), y(t-1), \ldots \} \quad (8)$$

The unknown parameters in $\theta$ are estimated with the recursive least-squares (RLS) algorithm (Ljung, 1999)

$$\hat{\theta}(t) = \hat{\theta}(t-1) + K(t)(y(t) - \varphi^\top(t-1)\hat{\theta}(t)) \quad (9a)$$

$$K(t) = P(t-1)\varphi(t-1) \times (\lambda + \varphi^\top(t-1)P(t-1)\varphi(t-1))^{-1} \quad (9b)$$

$$P(t) = (I - K(t)\varphi^\top(t-1))P(t-1)/\lambda \quad (9c)$$

where $\hat{\theta}(t)$ is a vector of parameter estimates, $K(t)$ is the injection gain, $\lambda$ is the forgetting factor, and $P(t)$ is a matrix of parameter estimate error covariances (when $\lambda = 1$, which is used in the remainder of this paper).

Based on the parameter estimate from the RLS algorithm, we introduce the process model or one-step-ahead predictor

$$\hat{y}(t+1 | t) = E\{ y(t+1) | \mathcal{Y}(t) \} = \varphi^\top(t)\hat{\theta}(t) \quad (10)$$

### 2.1 The Dual Control Objective

By setting $\mu = 0$ the objective of a dual controller can be stated as minimizing

$$J_\infty = \sum_{k=0}^{\infty} E \left\{ (y_{\text{ref}}(t+1+k) - y(t+1+k))^2 \mid \mathcal{Y}(t) \right\} \quad (11)$$

where $y_{\text{ref}}(t+1+k)$ is the output reference. That is, at time $t$ the objective is to minimize future output error $(y_{\text{ref}}(t+1+k) - y(t+1+k))^2$, $k = 0, 1, \ldots$, using the information gathered up to time $t$, $\mathcal{Y}(t)$. This is a fairly standard objective in optimal control; the main challenge comes from the lack of a model capable of predicting future outputs $y(t+1+k)$, $k = 0, 1, \ldots$. As noted by Feldbaum (1961), an optimal controller must in this case find the best tradeoff between control and excitation.

The objective function (11) can be approximated in a variety of ways. The simplest approximation is to use a one-step-ahead cost function and then replace $y(t+1)$ by the model $\hat{y}(t+1)$; this gives a simple certainty-equivalence controller. Minimization of a one-step-ahead objective without replacing $y(t+1)$ gives a cautious controller (meaning less aggressive control when parameter uncertainty is high). A probing effect appears when the horizon is of length 2 or more, meaning that in addition to moving the output toward $y_{\text{ref}}$ the control signal attempts to reduce parameter uncertainty through excitation or probing.

### 3. CONTROLLER

#### 3.1 Objective Reformulation

We first rewrite the objective function (11) as

$$J_\infty = E \left\{ (y_{\text{ref}}(t+1) - y(t+1))^2 \mid \mathcal{Y}(t) \right\}$$

$$+ \sum_{k=1}^{\infty} E \left\{ (y_{\text{ref}}(t+1+k) - y(t+1+k))^2 \mid \mathcal{Y}(t) \right\} \quad (12)$$

The first term is from here on referred to as $J_1$; that is,

$$J_1 = E \left\{ (y_{\text{ref}}(t+1) - y(t+1))^2 \mid \mathcal{Y}(t) \right\} \quad (13)$$

We now subtract and add the predictor (10) and get

$$J_1 = E \left\{ (y_{\text{ref}}(t+1) - \hat{y}(t+1))^2 \right\}$$

In order to simplify the notation, the condition in the expectation is no longer explicitly stated. The cost (14) can be expanded to

$$J_1 = E \left\{ (y_{\text{ref}}(t+1) - \hat{y}(t+1))^2 \right\}$$

$$- 2(y_{\text{ref}}(t+1) - \hat{y}(t+1))\left(\varphi^\top(t)\hat{\theta}(t) + v(t+1)\right)$$

$$+ (\varphi^\top(t)\hat{\theta}(t) - v(t+1))^2 \right\} \quad (15)$$

where $\hat{\theta}(t) = \theta - \hat{\theta}(t)$. The first of the three terms in (15) is deterministic; the second term is zero since both

$$E\left\{ \hat{\theta}(t) \right\} = E\left\{ \theta - \hat{\theta}(t) \right\} = \hat{\theta}(t) - \hat{\theta}(t) = 0 \quad (16)$$

and $E\{v(t+1)\} = 0$. Since $\hat{\theta}(t)$ and $v(t+1)$ are independent,

$$P(t) = E\left\{ \hat{\theta}(t)\hat{\theta}^\top(t) \right\}, \quad \text{and} \quad r = E\left\{ v(t+1)^2 \right\} \quad (17)$$

we can write the third term in (15) as

$$\varphi^\top(t)P(t)\varphi(t) + r \quad (18)$$

We then have that (15) can be written

$$J_1 = (y_{\text{ref}}(t+1) - \varphi^\top(t)\hat{\theta}(t))^2 + \varphi^\top(t)P(t)\varphi(t) + r \quad (19)$$

given $\mathcal{Y}(t)$. The reformulation of the first stage cost highlights the tradeoff between output control and caution, represented by the first and second terms in (19), respectively. That is, a large $P(t)$ matrix leads to a reward for small signals, meaning that a cautious input reduces the cost. Note that there are no stochastic variables in the one-step-ahead cost (19), which means that a myopic optimal controller can be found by minimizing $J_1$.

In order to obtain a probing effect, we use an identical cost function for the second stage time. That is, we set the total cost of the first two stages to

$$J_2 := \sum_{k=0}^{1} \left\{ (y_{\text{ref}}(t+1+k) - \varphi^\top(t+k)\hat{\theta}(t+k))^2 \right. \right.$$} 

$$+ \varphi^\top(t+k)P(t+k)\varphi(t+k) + r \right\} \quad (20)$$

The probing effect is achieved since a certain choice of input $u(t)$ will reduce the elements of the predicted future parameter estimate error covariance matrix $P(t+1)$ such that the cost is reduced. In other words, an input $u(t)$ that probes or excites the system in a manner that reveals information about the parameters and thereby reduces their estimate uncertainty is rewarded.

The infinite horizon objective function (11) can now be approximated in the following manner:
Another simplification of the cost function is obtained by truncating the horizon from infinity to some finite number $N > 1$. This simplified objective function can then be written

$$J_N \approx \sum_{k=0}^{N-1} \left\{ (y_{ref}(t+1+k) - \varphi^T(t+k)\hat{\theta}(t+k))^2 + \varphi^T(t+k)P(t+k)\varphi(t+k) + r \right\} + \sum_{k=N}^{\infty} E \left\{ (y_{ref}(t+1+k) - y(t+1+k))^2 \middle| \mathcal{Y}(t) \right\}$$

(21)

Since we apply a model predictive control strategy, the minimizer of the objective function (subject to the specified constraints) will be found at every time instant. The solution to the problem contains an open-loop optimal control sequence; the first input in this sequence is applied to the plant. Hence, it can be argued that the first time stage in the objective is the most important term.

For this reason, we approximate the second sum using the model or predictor (10) instead of the expected value of the output and call the approximated cost $V_N$. We also add a cost of input usage. This gives

$$J_N \approx V_N := \sum_{k=0}^{N-1} \left\{ (y_{ref}(t+1+k) - \varphi^T(t+k)\hat{\theta}(t+k))^2 + \varphi^T(t+k)P(t+k)\varphi(t+k) + r \right\} + \sum_{k=2}^{N-1} \left\{ w_2 (y_{ref}(t+1+k) - \hat{y}(t+1+k))^2 + w_3 u^2(t+k) \middle| \mathcal{Y}(t) \right\}$$

(23)

where $w_2$ and $w_3$ are cost weights. Note that

$$\hat{y}(t+1+k | t) = \varphi^T(t+k)\hat{\theta}(t)$$

(24)

meaning the output is predicted over the horizon using the current parameter estimate.

### 3.2 Constraints

A second contribution in our proposed algorithm is adding an adapted version of the recursive least-squares algorithm (9) as a set of equality constraints in the optimization problem. We have pursued similar approaches in Heirung et al. (2012) and Heirung et al. (2013), although to different ends. Adding these estimation equations gives the optimization solver a measure of how the input sequence affects parameter estimate uncertainty through predicting future covariances $P(k)$. Before discussing the addition of the modified estimation equations as constraints, we discuss the certainty-equivalence principle used in our controller, as well some standard MPC constraints.

Using the latest parameter estimate $\hat{\theta}(t)$, the output predictor equation (10) is added as a constraint at all future time instants $k$ in the prediction horizon as

$$\hat{y}(k+1 | t) = \varphi^T(k)\hat{\theta}(t), \quad k \in \{t, \ldots, t + N - 1\}$$

(25)

Since the regressor contains predicted future outputs when $k > t$, we use

$$\varphi^T(k) = [u(k), \ldots, u(k-n_b+1), -\hat{y}(k), \ldots, -\hat{y}(k-n_a+1)]^T$$

(26)

(cf. (6)) where

$$\hat{y}(i) := y(i)$$

if $i \leq t, \ i \in \{k-n_a+1, \ldots, k\}$

(27)

meaning $\hat{y}(i)$ is a recorded output (as opposed to a prediction) if $i$ corresponds to a current or past time instant.

For $k = t + 1$, we set $\hat{\varphi}(k-1) = \varphi(t)$, which contains past inputs and outputs and is passed from the simulator to the optimization problem. Note that in (27), $\hat{\theta}(t)$ is the most recent parameter estimate produced by the estimation algorithm (9) during simulation at time $t$. The predictor (27) is added as an equality constraint to the optimization problem. Since the output is predicted in this manner our controller becomes a certainty-equivalence type MPC. Finally, note that $\hat{\theta}(t)$ is a constant in the optimization problem solved at time $t$; this means that (27) is a linear equality constraint.

An MPC commonly includes bounds on inputs and outputs as inequality constraints (“box constraints”) as part of the online optimization problem. The input bounds are formulated

$$u_{min} \leq u(k) \leq u_{max}, \quad k \in \{t, \ldots, t + N - 1\}$$

(28)

The minimum and maximum values $u_{min}$ and $u_{max}$ are usually based on either hardware limitations or similar physical constraints. Since the model contains unknown parameters, the constraints on the output $y(t)$ cannot be directly included in the optimization problem. Instead, we simply add the bounds the predicted outputs:

$$y_{min} \leq \hat{y}(k) \leq y_{max}, \quad k \in \{t + 1, \ldots, t + N\}$$

(29)

There is no guarantee that predicted outputs $\hat{y}(k)$ being feasible with respect to the bounds implies that the actual output $y(t)$ would stay within the limits if some open-loop optimal input sequence $u^*(t), \ldots, u^*(t + N - 1)$ were implemented on the real process. However, since the parameter estimates improve over time, the box constraint (29) is more likely to ensure that

$$y_{min} \leq y(t) \leq y_{max}$$

We now add the necessary constraints for the optimization algorithm to have a way of predicting $P(t+1)$ based on its choice of $u(t)$. Since the future parameter estimates cannot be predicted (without future data), we add only equations (9b)–(9c) as equality constraints. We only need $P$ at $t+1$ and hence include the equations in the form

$$K(t+1) = P(t)\hat{\varphi}(t)(\lambda + \varphi^T(t)P(t)\hat{\varphi}(t))^{-1}$$

(30a)

$$P(t+1) = (I - K(t+1)\varphi^T(t))P(t)/\lambda$$

(30b)

### 3.3 The Online Optimization Problem

Based on the above discussion, we can now state the full optimization problem solved online at every time stage $t$: 

$$J_N \approx \sum_{k=0}^{N-1} \left\{ (y_{ref}(t+1+k) - \varphi^T(t+k)\hat{\theta}(t+k))^2 + \varphi^T(t+k)P(t+k)\varphi(t+k) + r \right\} + \sum_{k=N}^{\infty} E \left\{ (y_{ref}(t+1+k) - y(t+1+k))^2 \middle| \mathcal{Y}(t) \right\} + \sum_{k=2}^{N-1} \left\{ w_2 (y_{ref}(t+1+k) - y(t+1+k))^2 + w_3 u^2(t+k) \middle| \mathcal{Y}(t) \right\}$$

(21)
\[
\min_{u(k)} V_N = \sum_{k=0}^{N-1} \left\{ (y_{\text{ref}}(t+1+k) - \varphi^T(t+k)\hat{\theta}(t+k))^2 \\
+ \varphi^T(t+k)P(t+k)\varphi(t+k) + \ldots \right\} \\
+ \sum_{k=2}^{N} \left\{ w_2(y_{\text{ref}}(t+1+k) - \hat{y}(t+1+k))^2 \\
+ w_3u^2(t+k) \mid Y(t) \right\} \tag{31a}
\]

subject to
\[
\dot{y}(k \mid t) = \varphi^T(t-k)\hat{\theta}(t), \quad k = t+1, \ldots, t+ N \\
K(t+1) = P(t)\varphi(t)(\lambda + \varphi^T(t)P(t)\varphi(t))^{-1} \\
P(t+1) = (I - K(t+1)\varphi^T(t))P(t)/\lambda \\
u_{\text{min}} \leq u(k) \leq u_{\text{max}}, \quad k = t, \ldots, t + N - 1 \\
y_{\text{min}} \leq \hat{y}(k) \leq y_{\text{max}}, \quad k = t + 1, \ldots, t + N \tag{31b}
\]

An important consequence of the nonlinear equality constraints (31c)–(31d), which represent the estimation algorithm, is that the optimization problem becomes a non-convex nonlinear programming (NLP) problem.

The optimization problem has a number of variables fixed; the values of \(\varphi(t)\) (except for \(u(t)\)), \(\hat{\theta}(t)\), and \(P(t)\) are all passed from the simulation to the optimization algorithm and are fixed parameters in the optimization problem. Note that \(\varphi(t+1)\) being fixed implies that \(y(t), \ldots, y(t-n_a)\) and \(u(t), \ldots, u(t-n_b)\) are all given and fixed in the NLP problem. This means that the controller is neither output nor state feedback; rather, we have feedback from a hyperstate containing the current output \(y(t)\), the parameter estimates \(\hat{\theta}(t)\), the covariance matrix \(P(t)\), as well as a history of inputs and outputs.

4. IMPLEMENTATION

Our implementation of the algorithm is written in MATLAB and GAMS (GAMS Development Corporation, 2012) with IPOPT (Cervantes et al., 2000) as the NLP solver for the online optimization problem. IPOPT is a state-of-the-art open-source interior-nLP solver, which exploits the sparsity of the NLP and is capable of solving large-scale problems. The system is simulated in MATLAB; GAMS is called at every iteration and returns a (locally) open-loop optimal input sequence \(u^*(t), \ldots, u^*(t+N-1)\). The example runs on a standard laptop computer and the optimization problems are solved reasonably fast with solution times ranging from 0.21 to 1.04 CPUs. The implementation is not written with a focus on speed of execution.

5. APPLICATION EXAMPLE

As an application example we consider a tank containing a solution of salt dissolved in water. Let \(y(t)\) be the amount of salt in the tank, \(V(t)\) be the volume of solution in the tank, and \(c_{\text{out}}(t) = x(t)/V(t)\) be the concentration in the tank. A solution with constant known concentration \(c_{\text{in}}\) flows into the tank at a rate \(q_{\text{in}}(t) = k_1u(t)\) where \(u(t) \in [0,1]\) is a valve setting and \(k_1\) is an unknown constant. Solution flows out of the tank at a constant unknown rate \(q_{\text{out}}\) and with variable concentration \(c_{\text{out}}(t)\).

This system can be modeled with the linear first-order differential equation
\[
\dot{y} = -\frac{q_{\text{out}}}{V}y(t) + c_{\text{in}}k_1u(t) \tag{32}
\]
A sketch of the system is shown in Figure 1.

![Fig. 1. Sketch of the mixture example problem.](image)

We can discretize equation (32) to obtain a discrete-time equation of the form
\[
y(t) = -a_1y(t-1) + b_1u(t) \tag{33}
\]
We want to control the inlet flow rate to keep the concentration at \(y_{\text{ref}}\) while identifying the unknown parameters \((a_1\) and \(b_1\) in equation (33)).

In the numerical example we use the following parameter values: \(y_{\text{ref}} = 2.5, y(0) = 2.00, a_1 = -0.60, b_1 = 5.50, a_1(0) = -0.10, b_1(0) = 0.00, P(0) = 1 \times 10^2I, N = 6, w_2 = 1.00 \times 10^{-2}, w_3 = 1.00, u_{\text{min}} = 0, u_{\text{max}} = 1, y_{\text{min}} = 0, y_{\text{max}} = 10; the system is simulated for 10 time steps and no noise is used in the simulation.

The results are presented in Figure 2. The most significant result is that the Dual MPC generates a control input that probes or excites the system, even though the output error is very small at \(t = 0\). The probing leads to both parameter estimates converging in 2 time steps; the diagonal elements of the covariance matrix decrease accordingly. The output exhibits some oscillation due to the probing, but settles fairly quickly close to the reference of 2.5. The figure also contains results from applying a certainty-equivalence (CE) MPC to the same control problem. The initial input gain estimate is zero \((\hat{b}_1(0) = 0)\), which causes the CE MPC to generate a zero-input since the system appears uncontrollable. This again leads to a complete lack of information about the input gain and the situation is never resolved. This problem is avoided by the Dual MPC since it knows that excitation is necessary for good future performance.

6. CONCLUSIONS AND FUTURE WORK

We present a new model predictive controller exhibiting dual features in that it actively probes the system for information when the parameter estimates are poor. The main contribution is that the cost function is based on an exact reformulation of the output error for the first time step. This resulting expression is also used for the second stage cost and provides the probing reward in the cost
function. The objective function is evaluated by the NLP solver with the help of constraints based on a recursive least-squares algorithm for parameter estimation.

Future work includes extending the controller to larger systems with more unknown parameters as well as developing a version that can handle time-varying parameters. The algorithm will also be extended to multivariable systems. Analysis of stability and convergence properties will be investigated at a later point in time. We will also analyze the nonconvexity of the optimization problem and investigate the effect of finding the global solution.

The tradeoff between an update constraint for \( R(t + 1) \) with a more complicated objective function and the update constraint for \( P(t + 1) \) and a simpler objective function (done here) in the NLP will be the topic of a future paper.

REFERENCES


