



# Model Predictive Control with Integral Action: A simple MPC algorithm

David Di Ruscio

*Telemark University College, Kjolnes ring 56, 3918 Porsgrunn, Norway. E-mail contact: David.Di.Ruscio@hit.no*

---

## Abstract

A simple Model Predictive Control (MPC) algorithm of velocity (incremental) form is presented. The proposed MPC controller is insensitive to slowly varying system and measurement trends and therefore has integral action. The presented algorithm is illustrated by both simulations and practical experiments on a quadruple tank MIMO process.

*Keywords:* MIMO systems, model predictive control, optimal controller, integral action, constraints

---

## 1. Introduction

Model Predictive Control (MPC) algorithms are studied in a number of books (e.g. [Huang and Kadali \(2008\)](#), [Maciejowski \(2002\)](#), [Camacho and Bordons \(1995\)](#), [Bitmead et al. \(1990\)](#)) and in numerous papers, see e.g. [Maeder et al. \(2009\)](#) and the references therein.

MPC has its origin in the theory of optimal control problems, [Pontryagin et al. \(1956\)](#) and [Bellman \(1957\)](#), and in lecture notes on optimal control theory at MIT in the early 1960-ties which resulted in the book [Athans and Falb \(1966\)](#).

Parts of the optimal control theory are believed developed during the Apollo program in the early 60-ties where there was a great focus on minimum time (time optimal control) and minimum fuel (fuel optimal control) problems, see e.g. [Athans and Falb \(1966\)](#) Ch. 7 where the minimum principle by [Pontryagin et al. \(1956\)](#) is used to solve the optimal control problems. This also successful resulted in the state feedback optimal control law in the Apollo Lunar-Module Autopilot [Widnall \(1970\)](#).

The above mentioned optimal control problems are most often based on a fixed time optimization interval from say a constant initial time  $t_0 = 0$  to a final time, say  $L$ , i.e. a time optimal control problem where a cost

functional  $J(u) = \int_0^L dt = L$  and  $L$  is free, are optimized with respect to the control action  $u$  subject to a model and constraints. MPC problems are however based on a moving horizon (i.e. a receding horizon) strategy [Garca et al. \(1989\)](#) where we consider the initial time of the optimization interval to be equal to the present time  $t$ , and the final time of the optimization interval to be equal to the present time plus a prediction horizon, say  $t+L$  where  $L$  is the prediction horizon. There is no theoretical difference between standard optimal control problems and MPC problems and notice that the functional  $J(u) = \int_t^{t+L} dt = L$ .

An early description of the moving horizon or receding horizon optimal control strategy where the initial integration time is set equal to the present time,  $t$ , and the final horizon is set equal to  $t + L$  where  $L > 0$  is a constant time interval is as presented in [Propoi \(1963\)](#) and [Balchen et al. \(1970\)](#) p. 208 (in Norwegian).

An early survey paper on MPC is [Garca et al. \(1989\)](#). A survey of both linear and nonlinear MPC is given in, e.g. [Maeder et al. \(2009\)](#), [Qin and Badgwell \(2003\)](#). Some early work on non-linear MPC is presented in [Balchen et al. \(1992\)](#) and the references therein.

The Generalized Predictive Control (GPC) algorithm by [Clarke et al. \(1987\)](#) is an algorithm based on an input and output Controlled Auto Regressive and Integrated Moving Average (CARIMA) model. The

GPC algorithm gives integral action of the closed loop system and the present state is estimated using the model and some old known input and output data.

Standard MPC algorithms usually do not achieve integral action and there is one main reason for this. The answer is that integral action is not necessarily optimal. However, if integral action is wanted there is some common methods to achieve this.

One commonly used method is to augment an integrator at the input, i.e. augment the plant model with an integrator  $u_k = u_{k-1} + \Delta u_k$ . MPC algorithms is of state feedback type and in this case the MPC algorithm is a function of the plant state estimate  $\hat{x}_k$  and the estimate  $\hat{u}_{k-1}$ . The plant control is then  $u_k = u_{k-1} + \Delta u_k^*$  where  $\Delta u_k^*$  is the MPC calculated optimal control deviation. One should notice that in this case and due to unknown disturbances the actual previous control  $u_{k-1}$  is not necessarily equal to the estimate  $\hat{u}_{k-1}$ . This method is among others described in [Mayne et al. \(2000\)](#).

Another commonly used strategy to incorporate integral action is to augment an integrator model  $d_k = d_{k-1}$  of a constant disturbance  $d$  influencing the plant model state equation and the output equation. This strategy may be viewed as putting the integrator at the output. This strategy is also among others described in [Mayne et al. \(2000\)](#).

Recently in [Ruscio \(2012\)](#) a simple Linear Quadratic (LQ) optimal controller algorithm with integral action is presented. In this method constant or slowly varying disturbances in the state and output equations are removed from the problem by working on deviation models.

The contributions of this paper may be itemized as follows:

- In this paper an MPC algorithm with integral action, along the same lines as used in the LQ optimal controller with integral action method, in [Ruscio \(2012\)](#) is presented.

## 2. Problem formulation

Given a process model

$$x_{k+1} = Ax_k + Bu_k + v, \quad (1)$$

$$y_k = Dx_k + w, \quad (2)$$

where  $x_k \in \mathbb{R}^n$  is the state vector,  $u_k \in \mathbb{R}^r$  is the control input vector,  $y_k \in \mathbb{R}^m$  is the output (measurement) vector,  $A$ ,  $B$  and  $D$  are system matrices of appropriate dimensions, and  $x_0$  is the initial state.

The disturbances  $v$  and  $w$  may both be unknown, i.e.,  $v$  may be an unknown constant or slowly varying

process disturbance, and  $w$  may be an unknown constant or slowly varying measurements noise vector.  $v$  and  $w$  may represent trends or drifts.

Note that the variables  $u_k$  and  $y_k$  in the model eqs. (1) and (2) are the actual input and output variable, respectively. Furthermore, note that the model eqs. (1) and (2) may arise from linearizing non-linear models around some nominal steady state variables. The model may also rise from system identification based on trended input and output data. Hence, in these cases, the external noise variables  $v$  and  $w$  are known, but the resulting control algorithm to be presented in this paper is insensitive to these noise variables  $v$  and  $w$ . Furthermore the system and the measurements may be influenced by drifts and in these cases the noise variables  $v$  and  $w$  may be unknown and slowly varying. Hence, the model eqs. (1) and (2) is a realistic model.

We will study the MPC controller subject to the following performance index,

$$J_k = \sum_{i=1}^L ((y_{k+i} - r_{k+i})^T Q_i (y_{k+i} - r_{k+i}) + \Delta u_{k+i-1}^T P_i \Delta u_{k+i-1}), \quad (3)$$

where  $k$  is the present discrete time,  $\Delta u_k = u_k - u_{k-1}$  is the control rate of change (or control increment),  $r_k$  is a reference signal and  $Q_i$  and  $P_i$  are symmetric positive semi-definite weighting matrices of appropriate dimensions. For finite prediction horizon  $L$ , then  $Q_L$  may be chosen as the solution to Riccati equation of the problem to ensure closed loop nominal stability.

The above MPC objective criterion may be written in more compact form as

$$J_k = (y_{k+1|L} - r_{k+1|L})^T Q (y_{k+1|L} - r_{k+1|L}) + \Delta u_{k|L}^T P \Delta u_{k|L}, \quad (4)$$

where  $Q \in \mathbb{R}^{Lm \times Lm}$  is a block diagonal matrix with  $Q_i \forall i = 1, \dots, L$  on the block diagonal.  $P \in \mathbb{R}^{Lr \times Lr}$  is defined similar with  $P_i \forall i = 1, \dots, L$  on the block diagonal. The notation used to define the vectors in eq. (4) is defined in [Appendix A](#).

In this paper we consider input rate of change and amplitude constraints. These constraints may be formulated as a linear inequality,

$$\mathcal{A} \Delta u_{k|L} \leq b_k, \quad (5)$$

where the matrix  $\mathcal{A}$  and the vector  $b_k$  are defined later in [Sec. 3.2](#).

The MPC problem is now equivalent to a Quadratic Programming (QP) problem, i.e., the objective function eq. (4) with the process model, eqs (1) and (2), is minimized with respect to the unknown vector of future control increments, subject to the process constraints eq. (5), i.e.,

$$\Delta u_{k|L}^* = \arg \min_{\mathcal{A} \Delta u_{k|L} \leq b_k} J_k. \quad (6)$$

The simplified MPC strategy of including a control horizon,  $1 \leq L_u \leq L$ , and instead calculating a reduced number of future controls  $\Delta u_{k|L_u}$ , will be discussed and solved later in the paper.

## 2.1. Model discussion

Note that the model eqs. (1) and (2), when  $v$  and  $w$  are constant vectors, is not unique. The constant trends  $v$  and  $w$  may be incorporated in the model by including one additional state. We find that the following model is equivalent

$$\begin{bmatrix} \bar{x}_{k+1} \\ x_{k+1} \\ z_{k+1} \end{bmatrix} = \begin{bmatrix} \bar{A} \\ A & v \\ 0_{1 \times n} & 1 \end{bmatrix} \begin{bmatrix} \bar{x}_k \\ x_k \\ z_k \end{bmatrix} + \begin{bmatrix} \bar{B} \\ B \\ 0 \end{bmatrix} u_k, \quad (7)$$

$$y_k = \begin{bmatrix} \bar{D} \\ D & w \end{bmatrix} \begin{bmatrix} \bar{x}_k \\ x_k \\ z_k \end{bmatrix}, \quad (8)$$

with the initial state vector as

$$\bar{x}_0 = \begin{bmatrix} x_0 \\ z_0 \end{bmatrix}, \quad (9)$$

and with  $z_0 = 1$  in order to take care of the constant trends.

## 3. MPC algorithm

### 3.1. Problem solution

In order to solve the MPC optimal control problem eq. (6) we need a model which is independent of the unknown disturbances. For the sake of generality we are focusing on state space modeling.

From the state eq. (1) we have

$$\Delta x_{k+1} = A \Delta x_k + B \Delta u_k, \quad (10)$$

where  $\Delta x_k = x_k - x_{k-1}$ . From the measurement eq. (2) we have

$$y_k = y_{k-1} + D \Delta x_k. \quad (11)$$

Augmenting eqs. (10) with (11) gives the state space model

$$\begin{bmatrix} \bar{x}_{k+1} \\ \Delta x_{k+1} \\ y_k \end{bmatrix} = \begin{bmatrix} \bar{A} \\ A & 0_{n \times m} \\ D & I_{m \times m} \end{bmatrix} \begin{bmatrix} \bar{x}_k \\ \Delta x_k \\ y_{k-1} \end{bmatrix} + \begin{bmatrix} \bar{B} \\ B \\ 0_{m \times r} \end{bmatrix} \Delta u_k, \quad (12)$$

$$y_k = \begin{bmatrix} \bar{D} \\ D & I_{m \times m} \end{bmatrix} \begin{bmatrix} \bar{x}_k \\ \Delta x_k \\ y_{k-1} \end{bmatrix}. \quad (13)$$

Hence, we have a strictly proper state space model of the form

$$\tilde{x}_{k+1} = \tilde{A} \tilde{x}_k + \tilde{B} \Delta u_k, \quad (14)$$

$$y_k = \tilde{D} \tilde{x}_k. \quad (15)$$

The state space model eqs (12), (13) (or equivalently (14), (15)) may be used to define a Prediction Model (PM) of the form

$$y_{k+1|L} = p_L + F_L \Delta u_{k|L}, \quad (16)$$

where

$$p_L = O_L \tilde{A} \tilde{x}_k, \quad (17)$$

$$F_L = \begin{bmatrix} O_L \tilde{B} & H_L^d \end{bmatrix}, \quad (18)$$

and where  $O_L$  is the extended observability matrix of the pair  $\tilde{A}, \tilde{D}$  and  $H_L^d \in \mathbb{R}^{mL \times (L-1)r}$  the Toeplitz matrix of impulse response matrices  $\tilde{D} \tilde{A}^{i-1} \tilde{B} \in \mathbb{R}^{m \times r}$ . See Appendix A for definitions.

The performance index eq. (4) with the PM eq. (16) can be written as a quadratic function on standard form, i.e.,

$$J_k = \Delta u_{k|L}^T H \Delta u_{k|L} + 2 f_k^T \Delta u_{k|L} + J_0, \quad (19)$$

where

$$H = F_L^T Q F_L + P, \quad (20)$$

$$f_k = F_L^T Q (p_L - r_{k+1|L}). \quad (21)$$

The constant term  $J_0$  in eq. (19) is not a function of the unknown  $\Delta u_{k|L}$  and then not needed and therefore not presented.

Notice that when the constraints in the MPC problem eq. (5) is inactive, then the unconstrained MPC controls is given by

$$\Delta u_{k|L}^* = -H^{-1} f_k. \quad (22)$$

Usually we have process constraints and this will be discussed in the next Sec. 3.2.

### 3.2. Constraints

It make sense to specify input rate of change constraints, i.e.

$$\Delta u_{\min} \leq \Delta u_{k|L} \leq \Delta u_{\max}, \quad (23)$$

and input amplitude constraints, i.e.

$$u_{\min} \leq u_{k|L} \leq u_{\max}. \quad (24)$$

Using the relationship

$$u_{k|L} = S \Delta u_{k|L} + c u_{k-1}, \quad (25)$$

where  $S \in \mathbb{R}^{rL \times rL}$  and  $c \in \mathbb{R}^{rL \times r}$  are matrices with ones and zeroes as defined in the Appendix A.

We find that the constraints may be written as the linear matrix inequality

$$\mathcal{A}\Delta u_{k|L} \leq b_k, \quad (26)$$

where

$$\mathcal{A} := \begin{bmatrix} I_{(rL \times rL)} \\ -I_{(rL \times rL)} \\ S \\ -S \end{bmatrix}, \quad (27)$$

and

$$b_k := \begin{bmatrix} \Delta u_{\max} \\ -\Delta u_{\min} \\ u_{\max} - cu_{k-1} \\ -u_{\min} + cu_{k-1} \end{bmatrix}. \quad (28)$$

The MPC algorithm is then to minimize the objective eq. (19) with respect to the constraints given by the linear inequality eq. (26).

This is a standard Quadratic Programming (QP) problem in terms of the unknown future control increments and the optimal solution is  $\Delta u_{k|L}^*$ , as defined in eq. (6). A receding horizon strategy is used and only the first control increment  $\Delta u_k^*$  in the calculated  $\Delta u_{k|L}^*$ , is used for control. The actual control action to the process is then  $u_k = \Delta u_k^* + u_{k-1}$ .

## 4. Reducing the number of unknowns future control actions

Usually when presenting MPC algorithms a control horizon is defined, and this control horizon is usually less than the prediction horizon  $L$  in order to reduce the number of unknown and then reducing the computation time.

### 4.1. Computing only the present control action

Consider now the extreme case in which the future control actions are equal to the present control action, i.e. such that  $u_{k+i-1} = u_k \forall i = 1, 2, \dots$ . In this case the only unknown control action is  $u_k$ , and equivalently  $\Delta u_k$ .

In this case we have that  $\Delta u_{k+i} = 0 \forall i = 1, 2, \dots$ . This gives

$$\Delta u_{k|L}^T P \Delta u_{k|L} = \Delta u_k P_1 \Delta u_k, \quad (29)$$

and hence we have the control objective

$$J_k = (y_{k+1|L} - r_{k+1|L})^T Q (y_{k+1|L} - r_{k+1|L}) + \Delta u_k^T P_1 \Delta u_k. \quad (30)$$

In this case we furthermore have a simple PM of the form

$$y_{k+1|L} = p_L + F \Delta u_k, \quad (31)$$

where

$$F = O_L \tilde{B}, \quad (32)$$

and where  $O_L$  is the extended observability matrix of the pair  $(\tilde{D}, \tilde{A})$ . The term  $p_L$  is unchanged and given by eq. (17).

This gives the control objective as a function of the increment  $\Delta u_k$  only, i.e. as follows

$$J_k = \Delta u_k^T H \Delta u_k + 2f_k^T \Delta u_k + J_0, \quad (33)$$

where

$$H := F^T Q F + P_1, \quad (34)$$

$$f_k := F^T Q (p_L - r_{k+1|L}). \quad (35)$$

In this case we find that the constraints may be written as the linear matrix inequality

$$\mathcal{A}\Delta u_k \leq b_k, \quad (36)$$

where

$$\mathcal{A} := \begin{bmatrix} I_{r \times r} \\ -I_{r \times r} \\ I_{r \times r} \\ -I_{r \times r} \end{bmatrix}, \quad (37)$$

and

$$b_k := \begin{bmatrix} \Delta u_{\max} \\ -\Delta u_{\min} \\ u_{\max} - u_{k-1} \\ -u_{\min} + u_{k-1} \end{bmatrix}. \quad (38)$$

The MPC algorithm is then to minimize the objective eq. (33) with respect to the constraints given by the linear inequality eq. (36). This is a standard quadratic programming problem in terms of the unknown and the optimal minimizing solution is  $\Delta u_k^*$ . A receding horizon strategy is used. The actual control action to the process is,  $u_k = \Delta u_k^* + u_{k-1}$ .

The strategy presented in this section is considerably reducing the computational time of the MPC algorithm. This strategy is demonstrated to work considerably well for the control of the quadruple tank process.

### 4.2. Including a control horizon

In order to reduce the number of unknown input variables it is common to include a control horizon, say

$L_u$  where  $1 \leq L_u \leq L$ . In this section we study the objective

$$J_k = \frac{1}{2} \sum_{i=1}^L ((y_{k+i} - r_{k+i})^T Q_i (y_{k+i} - r_{k+i}) + \sum_{i=1}^{L_u} \Delta u_{k+i-1}^T P_i \Delta u_{k+i-1}), \quad (39)$$

Hence, the compact form of this objective can be written as

$$J_k = (y_{k+1|L} - r_{k+1|L})^T Q (y_{k+1|L} - r_{k+1|L}) + \Delta u_{k|L_u}^T P_{L_u} \Delta u_{k|L_u}, \quad (40)$$

where  $P_{L_u} = P(1:rL_u, 1:rL_u)$ .

In this case we find a PM of the form as in eq. (31) with

$$F := F_L(:, 1:rL_u), \quad (41)$$

where  $F_L$  is defined in eq. (18). We are using MATLAB notation in eq. (41) and hence  $F$  is defined from all rows in  $F_L$  and the first  $rL_u$  columns.

Substituting this PM into the objective eq. (40) gives

$$J_k = \Delta u_{k|L_u}^T H \Delta u_{k|L_u} + 2f_k^T \Delta u_{k|L_u} + J_0, \quad (42)$$

where

$$H := F^T Q F + P_{L_u}, \quad (43)$$

$$f_k := F^T Q (p_L - r_{k+1|L}). \quad (44)$$

The constraints are then formulated as the linear inequality

$$\mathcal{A} \Delta u_{k|L_u} \leq b_k, \quad (45)$$

where in this case, for  $1 \leq L_u \leq L$  we have

$$\mathcal{A} := \begin{bmatrix} I_{(rL_u \times rL_u)} \\ -I_{(rL_u \times rL_u)} \\ S(1:rL_u, 1:rL_u) \\ -S(1:rL_u, 1:rL_u) \end{bmatrix}, \quad (46)$$

and

$$b_k := \begin{bmatrix} \Delta u_{\max}(1:rL_u) \\ -\Delta u_{\min}(1:rL_u) \\ u_{\max}(1:rL_u) - cu_{k-1} \\ -u_{\min}(1:rL_u) + cu_{k-1} \end{bmatrix}. \quad (47)$$

Hence we have a quadratic programming problem in the unknown vector  $\Delta u_{k|L_u} \in \mathbb{R}^{rL_u}$  of future control actions and only the first vector,  $\Delta u_k$  is used.

## 5. Constant references and large prediction horizon

We will here discuss a special case which leads to a particular simple solution to the optimal control problem. Consider the case where the references are constant, i.e., and that the prediction horizon is large or infinite. Then we may use the LQ index

$$\begin{aligned} J_i &= \frac{1}{2} \sum_{k=i}^{\infty} ((y_k - r)^T Q (y_k - r) + \Delta u_k^T P \Delta u_k) \\ &= \frac{1}{2} \sum_{k=i}^{\infty} (\tilde{y}_k^T Q \tilde{y}_k + \Delta u_k^T P \Delta u_k), \end{aligned} \quad (48)$$

where we have defined  $\tilde{y}_k = y_k - r$ .

If  $r$  is a non-zero constant reference then the measurements eq. (11) can be written as

$$y_k - r = y_{k-1} - r + D \Delta x_k. \quad (49)$$

The state and output eqs. (12) and (13) can then be rewritten as

$$\begin{bmatrix} \tilde{x}_{k+1} \\ \Delta x_{k+1} \\ y_k - r \end{bmatrix} = \begin{bmatrix} \tilde{A} \\ A & 0_{n \times m} \\ D & I_{m \times m} \end{bmatrix} \begin{bmatrix} \tilde{x}_k \\ \Delta x_k \\ y_{k-1} - r \end{bmatrix} + \begin{bmatrix} \tilde{B} \\ B \\ 0_{m \times r} \end{bmatrix} \Delta u_k, \quad (50)$$

$$\begin{bmatrix} \tilde{y}_k \\ y_k - r \end{bmatrix} = \begin{bmatrix} \tilde{D} \\ D & I_{m \times m} \end{bmatrix} \begin{bmatrix} \tilde{x}_k \\ \Delta x_k \\ y_{k-1} - r \end{bmatrix}. \quad (51)$$

Hence, we have a strictly proper state space model of the form

$$\tilde{x}_{k+1} = \tilde{A} \tilde{x}_k + \tilde{B} \Delta u_k, \quad (52)$$

$$\tilde{y}_k = \tilde{D} \tilde{x}_k \quad (53)$$

The state space model (50) and (51) (or equivalently (52) and (53)) with the performance index (48) defines a standard LQ optimal control problem. The optimal control is of the form

$$\Delta u_k = [G_1 \quad G_2] \begin{bmatrix} \Delta x_k \\ y_{k-1} - r \end{bmatrix}, \quad (54)$$

which can be rewritten as

$$u_k = u_{k-1} + G_1 \Delta x_k + G_2 (y_{k-1} - r). \quad (55)$$

The LQ optimal controller (55) gives  $y = r$  in steady state since the closed loop system is stable due to the properties of the LQ optimal controller. The states are seldom measured in practice. In this case we can use a state observer to define the deviation state  $\Delta x_k$ . However, another solution is to define  $\Delta x_k$  in terms of some past and known outputs  $\dots, y_{k-1}, y_k$  and some known inputs  $\dots, u_{k-1}$  and the model matrices  $A, B$  and  $D$ .

## 6. Numerical examples

### Example 6.1 (4 tank process (simulations))

Consider the quadruple tank process, [Johansson \(2000\)](#), with the non-linear state space model derived from mass balances and Bernulli's/Torricelli's law. By equating the potential energy and kinetic energy, i.e.  $mgh = \frac{1}{2}mv^2$  and solving for the velocity we obtain  $v = \sqrt{2gh}$ . Multiplying with the area,  $a$ , of the outlet hole of the tank we obtain the volumetric flow-rate,  $q$ , out of the tank as  $q = av = a\sqrt{2gh}$ .

Hence, a mass balance of the four tank process gives the state space model

$$A_1\dot{x}_1 = -a_1\sqrt{2gx_1} + a_3\sqrt{2gx_3} + \gamma_1k_1u_1, \quad (56)$$

$$A_2\dot{x}_2 = -a_2\sqrt{2gx_2} + a_4\sqrt{2gx_4} + \gamma_2k_2u_2, \quad (57)$$

$$A_3\dot{x}_3 = -a_3\sqrt{2gx_3} + (1 - \gamma_2)k_2u_2, \quad (58)$$

$$A_4\dot{x}_4 = -a_4\sqrt{2gx_4} + (1 - \gamma_1)k_1u_1, \quad (59)$$

where  $A_i \forall i = 1, \dots, 4$  is the cross-section area of tank  $i$ ,  $a_i \forall i = 1, \dots, 4$  is the cross-section area of the outlet pipe of tank  $i$ .

The flow  $k_1u_1$  from pump 1 may be divided into a flow  $\gamma_1k_1u_1$  into tank 1 and a flow  $(1 - \gamma_1)k_1u_1$  into tank 4, i.e. such that the flow from pump number 1 is  $k_1u_1 = \gamma_1k_1u_1 + (1 - \gamma_1)k_1u_1$ . Similarly, the flow  $k_2u_2$  from the second pump may be divided into a flow  $\gamma_2k_2u_2$  into tank 2 and a flow  $(1 - \gamma_2)k_2u_2$  into tank 3. Here  $\gamma_1$  and  $\gamma_2$  are fixed parameters. The system is non-minimum phase when choosing these parameters such that,  $0 < \gamma_1 + \gamma_2 < 1$ , and the system is minimum phase when,  $1 < \gamma_1 + \gamma_2 < 2$ . The numerical values for the above parameters, as well as nominal values for the states and control inputs, are chosen as presented in [Johansson \(2000\)](#).

The 4 tank process is studied in a number of papers, see e.g. [Gatzke et al. \(2000\)](#) where Internal Model Control (IMC) and Dynamic Matrix Control (DMC) were used. Here we use the proposed MPC controller with integral action as presented in [Sec. 3.1](#).

The results after using the MPC controller in [Sec. 3](#) in order to control the non-linear model eqs. (56)-(59) are presented in [Figures 1 and 2](#). The MATLAB `quadprog.m` function is used to solve the QP problem as described in [Sec. 3.2](#). The weighting matrices were chosen simply as  $P = I_2$  and  $Q = 0.0001I_2$ . Only the minimum phase case is illustrated.

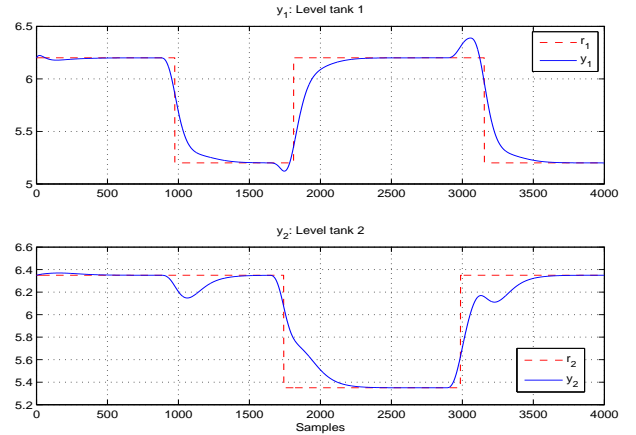


Figure 1: Simulation of the quadruple tank process and the minimum phase case in Example 6.1 with MPC control with integral action. Level references and actual levels illustrated. Prediction horizon  $L = 100$  and control horizon  $L_u = L$ .

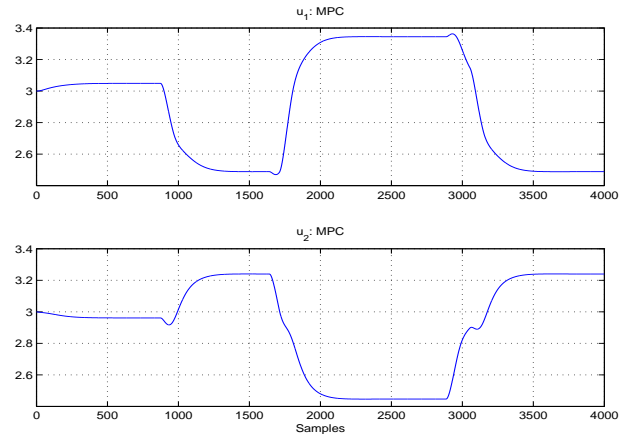


Figure 2: Simulation of the quadruple tank process and the minimum phase case in Example 6.1 with MPC control with integral action. Control inputs illustrated. Prediction horizon  $L = 100$  and control horizon  $L_u = L$ .

## 7. Experimental results on a quadruple tank process

A practical experiment with the MPC algorithm were performed on a quadruple tank process. The quadruple tank process is further discussed in Example 6.1.

The sampling rate in all experiments is one second. We started with empty tanks in all experiments. Hence, this may be viewed as a test for robustness for unknown non-linearities when using the proposed MPC controller. The quadruple tank process setup results in a non-minimum phase behavior.

The experiments are described in the following items.

1. The non-linear model, Eqs. (56)-(59) with measurements of the levels  $h_1$  and  $h_2$  in the lower tanks, were used in an Extended Kalman Filter (EKF) to estimate the four states in vector  $x$  needed in the MPC algorithm in order to formulate the term  $p_L$  in the PM.

The experiment is illustrated in Fig. (3). In order to reduce the computational time using LabView the simple MPC algorithm in Sec 4 were used. From Fig. (3) we see that this MPC algorithm performs well.

2. An open loop input experiment is designed as illustrated in Fig. (4) and the corresponding outputs, i.e. the levels in the two lower tanks, also illustrated in Fig. (5).
3. The input and output data are collected into data matrices  $U \in \mathbb{R}^{N \times 2}$ , and  $Y \in \mathbb{R}^{N \times 2}$  where the number of samples is  $N = 5459$ . The first  $N_{ID} = 4000$  first samples were used for identification. Hence, the last 1459 samples may be used for validation of the identified state space models. The data was also centered before use in the identification methods.
4. A First Principles (FP) model, very similar to the one presented in Example 6.1, were fitted to the process as well as believed possible. Using the input experiment as illustrated in Fig. (4) gave the simulated outputs as illustrated in Fig. (5). The Prediction Error (PE) criterion evaluated for the validation data was  $V_{fp} = 7.57$ .
5. The MATLAB IDENT Toolbox system identification function `pem.m` were used to identify a  $n = 4$  order state space model. The simulated outputs are illustrated in Fig. (5). The PE criterion evaluated for the validation data was  $V_{PEM} = 3.38$ .

6. The subspace system identification method, Di Ruscio (1996), Di Ruscio (2009) were used. The best DSR model with  $n = 4$  states was found with parameters  $L = 2$  and  $J = 29$ . The simulated outputs are illustrated in Fig. (5). The PE criterion evaluated for the validation data was  $V_{DSR} = 3.07$ .

7. Two SISO PI controllers were tuned by using the model based tuning method in Ruscio (2010). The model used was the DSR model. The experimental results using this decentralized control strategy is illustrated in Figs. (7) and (8).

8. The LQ optimal control strategy eq. (19) was implemented. The Kalman filter identified by the DSR method were used to identify the present state deviation  $\Delta x_k = x_k - x_{k-1}$  needed in the controller. The experimental results using this LQ optimal controller with integral action strategy is illustrated in Figs. (7) and (8).

The conclusions drawn from these experimental results are discussed in the following.

Interestingly the identified state space models, both from PEM and DSR, fits the real data better than the FP model. Here the simulated output, i.e. the behavior from the input  $u$ , to the output  $y$ , is used in order to calculate the PE criterion. The results using the FP model, the PEM model and the DSR model are  $V_{fp} = 7.57$ ,  $V_{PEM} = 3.38$  and  $V_{DSR} = 3.07$ , respectively. Interestingly the DSR model is slightly better to fit the validation data compared to the PEM model.

Based on this conclusion we are using the identified DSR model for both tuning the PI controllers and for use in the LQ optimal controller with integral action strategy eq. (20). The deterministic part of the model, i.e.  $x_{k+1} = Ax_k + Bu_k$  and  $y_k = Dx_k$ , were used to tune the PI controller strategy (by first using the RGA pairing strategy, Bristol (1966)), as well as for the calculation of the feedback matrices  $G_1$  and  $G_2$ . Furthermore the DSR identified Kalman filter gain matrix  $K$  were used in the Kalman filter on deviation form as presented in Ruscio (2012), for estimating the deviation states  $\Delta x_k$  needed in eq. (20).

As we see from Figs. (7) and (8) the LQ strategy works very well compared to the PI controller strategy. This is justified by comparing the Integrated Absolute (IAE) indices. The DSR model gave IAE indices 1.6849 and 1.3290 for level one and two, respectively, and for the PI controllers 2.2723 and 2.5141 for level one and two, respectively. It is also worth mentioning that it is very difficult to tune PI controllers for this process due to the non-minimum phase behavior of the process.

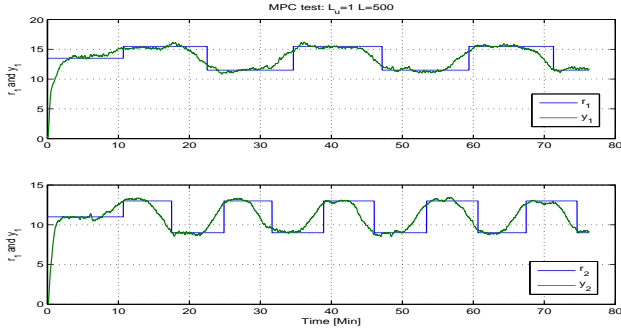


Figure 3: Practical run of the quadruple tank process as in Sec. 7, and the non-minimum phase case in Example 6.1 with MPC control with integral action. A control horizon  $L_u = 1$  and the algorithm in Sec. 4.

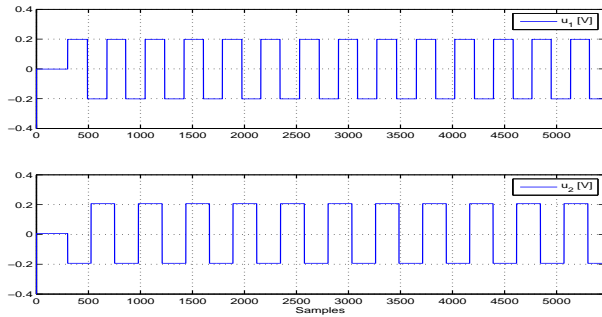


Figure 4: Open loop system identification input experiment, i.e. the volt input to the pumps.

## 8. Discussion

The presented MPC algorithm is based on a state space model of the plant and is therefore flexible to be used for MIMO systems. The algorithm may be combined with any state observer for estimating the present state, e.g. the Kalman filter, Jazwinski (1989) or simply with a state observer based on past inputs and outputs as described in Ruscio and Foss (1998).

The algorithm as presented in this paper is believed to work very similar as the GPC algorithm in Clarke et al. (1987). However, as mentioned in the introduction Sec. 1 the GPC algorithm is based on the use of an input and output CARIMA model. Such models are practical only for SISO systems. CARIMA models are capable of removing the influence of constant disturbances as in the state space model description in Eqs. (1) and (2) (SISO systems assumed).

The PM used in the GPC algorithm may be written on the form as in Eqs. (16) - (18). See e.g. Bitmead et al. (1990). One difference between the presented MPC algorithm and the GPC algorithm, is that the state estimate in the GPC algorithm, e.g. as in Eq. (17) is calculated based on the smallest number of past inputs and outputs. The presented MPC algorithm is

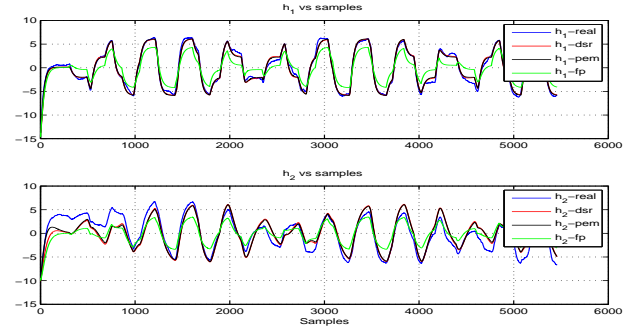


Figure 5: This figure illustrates the real measurements of the level in the two lower tanks as well as the corresponding simulated outputs of the system identification models, from DSR and PEM, as well as the simulated outputs from the first principles model.

more flexible with respect to state observers to be used. See e.g. Ruscio and Foss (1998) for a state observer along these lines.

## 9. Concluding remarks

A simple state space MPC controller with integral action on velocity (incremental) form for MIMO systems is presented.

## Acknowledgment

The author acknowledges the assistance of Mr. Danuska who did the practical implementation on the quadruple tank process.

## A. Notations used

The special structure of a Hankel matrix as well as some matching notations, which are frequently used throughout the paper, are defined in the following.

Given a vector

$$s_k \in \mathbb{R}^{nr} \quad \forall k = 0, 1, 2, \dots, \quad (60)$$

where  $nr$  is the number of rows in  $s_k$ .

Define integer numbers  $j$  and  $i$  and define the vector  $s_{j|i} \in \mathbb{R}^{nr}$  as follows

$$s_{j|i} \stackrel{def}{=} \begin{bmatrix} s_j \\ s_{j+1} \\ \vdots \\ s_{j+i-1} \end{bmatrix},$$

which is defined as an extended vector.



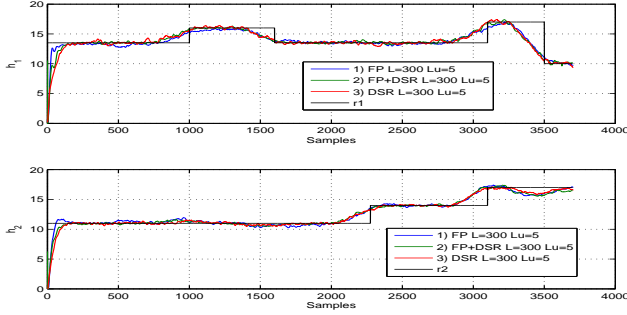


Figure 6: Quadruple tank process. Level in tank one upper and tank two lower. Illustrating the reference and the outputs from the process controlled by the proposed MPC controller, for three different cases. Case 1: FP model used in EKF to estimate the present state and linearized FP model in the MPC. Case 2: FP model used in EKF to estimate the present state and identified DSR model in the MPC. Case 3: DSR model and the corresponding Kalman filter used to estimate the present state and DSR model in the MPC.

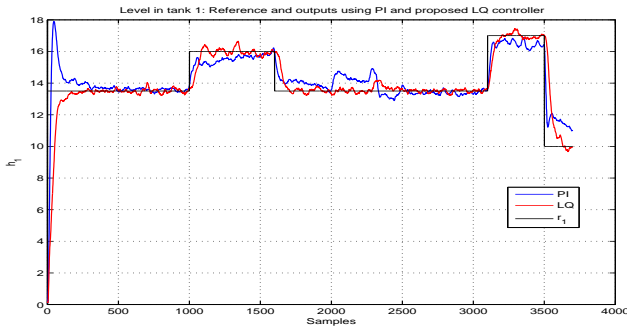


Figure 7: Quadruple tank process. Level in tank one. Illustrating the reference and the outputs from the process controlled by two single loop PI controllers, and the proposed LQ optimal controller with integral action. The LQ controller were constructed by using the DSR method for system identification. The DSR model was used to identify a Kalman filter for the system. The states were estimated with this Kalman filter and the deterministic part of the model were used to design the controller.

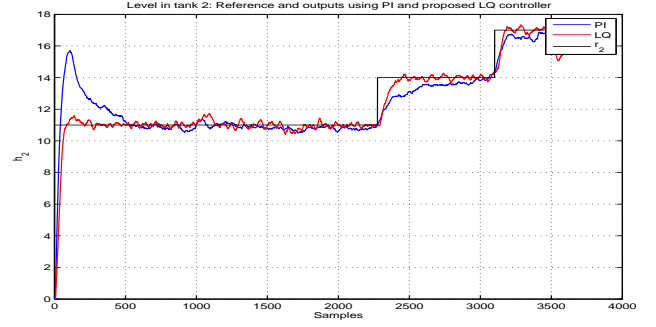


Figure 8: Quadruple tank process. Level in tank one. Illustrating the reference and the outputs from the process controlled by two single loop PI controllers, and the proposed LQ optimal controller with integral action. The LQ controller where constructed by using the DSR method for system identification. The DSR model was used to identify a Kalman filter for the system. The states were estimated with this Kalman filter and the deterministic part of the model were used to design the controller.

The integer numbers  $j$  and  $i$  have the following interpretations:

- $j$  start index or initial time in the sequence used to form  $s_{j|i}$ , i.e.,  $s_j$ , is the upper vector element in the extended vector  $s_{j|i}$ .
- $i$  is the number of  $nr$ -rows in  $s_{j|i}$ .

Examples of such vector processes,  $s_k$ , to be used in the above definition, are the measured process outputs,  $y_k \in \mathbb{R}^m$ , inputs,  $u_k \in \mathbb{R}^r$  and references,  $r_k \in \mathbb{R}^m$ .

The *extended observability matrix*,  $O_i$ , for the pair  $(D, A)$  is defined as

$$O_i \stackrel{\text{def}}{=} \begin{bmatrix} D \\ DA \\ \vdots \\ DA^{i-1} \end{bmatrix} \in \mathbb{R}^{im \times n}, \quad (61)$$

where the subscript  $i$  denotes the number of block rows.

The *reversed extended controllability matrix*,  $C_i^d$ , for the pair  $(A, B)$  is defined as

$$C_i^d \stackrel{\text{def}}{=} [ A^{i-1}B \quad A^{i-2}B \quad \dots \quad B ] \in \mathbb{R}^{n \times ir}, \quad (62)$$

where the subscript  $i$  denotes the number of block columns.

The lower block triangular Toeplitz matrix,  $H_i^d \in \mathbb{R}^{im \times (i+g-1)r}$ , for the quadruple matrices  $(D, A, B, E)$ .

$$H_i^d \stackrel{\text{def}}{=} \begin{bmatrix} E & 0_{m \times r} & 0_{m \times r} & \cdots & 0_{m \times r} \\ DB & E & 0_{m \times r} & \cdots & 0_{m \times r} \\ DAB & DB & E & \cdots & 0_{m \times r} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ DA^{i-2}B & DA^{i-3}B & DA^{i-4}B & \cdots & E \end{bmatrix}$$

where the subscript  $i$  denotes the number of block rows and  $i+g-1$  is the number of block columns, and where  $0_{m \times r}$  denotes the  $m \times r$  matrix with zeroes.

Define  $\Delta u_k = u_k - u_{k-1}$ . Using that  $u_k = \Delta u_k + u_{k-1}$ ,  $u_{k+1} = \Delta u_{k+1} + u_k = \Delta u_{k+1} + \Delta u_k + u_{k-1}$  and so on, we have

$$u_{k|L} = S\Delta u_{k|L} + cu_{k-1} \quad (63)$$

where  $S \in \mathbb{R}^{Lr \times Lr}$  and  $c \in \mathbb{R}^{Lr \times r}$  are given by

$$S = \begin{bmatrix} I_r & 0_r & \cdots & 0_r \\ I_r & I_r & \cdots & 0_r \\ \vdots & \vdots & \ddots & \vdots \\ I_r & I_r & \cdots & I_r \end{bmatrix}, \quad c = \begin{bmatrix} I_r \\ I_r \\ \vdots \\ I_r \end{bmatrix}, \quad (64)$$

where  $I_r$  is the  $r \times r$  identity matrix and  $0_r$  is the  $r \times r$  matrix of zeroes.

## References

- Athans, M. and Falb, P. L. *Optimal Control: An Introduction to the Theory and its Applications*. McGraw-Hill, 1966.
- Balchen, J. G., Fjeld, M., and Solheim, O. A. *Reguleringsteknikk Bind 3*. Tapir, 1970.
- Balchen, J. G., Ljungqvist, D., and Strand, S. State-space predictive control. *Modeling, Identification and Control*, 1992. 13(2):77–112. doi:10.4173/mic.1992.2.2.
- Bellman, R. *Dynamic Programming*. Princeton University Press, 1957.
- Bitmead, R. R., Gevers, M., and Wertz, V. *Adaptive Optimal Control: The Thinking Man's GPC*. Prentice-Hall, 1990.
- Bristol, E. H. On a new measure of interactions for multivariable process control. *Transactions on Automatic Control*, 1966. 11(11):113–114. doi:10.1109/TAC.1966.1098266.
- Camacho, E. F. and Bordons, C. *Model Predictive Control in the Process Industry*. Springer-Verlag, Berlin, 1995.
- Clarke, D. W., Mohtadi, C., and Tuffs, P. S. Generalized Predictive Control-Part I. *Automatica*, 1987. 23(2):137–148. doi:10.1016/0005-1098(87)90087-2.
- Di Ruscio, D. Combined Deterministic and Stochastic System Identification and Realization: DSR - A Subspace Approach Based on Observations. *Modeling, Identification and Control*, 1996. 17(3):193–230. doi:10.4173/mic.1996.3.3.
- Di Ruscio, D. Closed and Open Loop Subspace System Identification of the Kalman Filter. *Modeling, Identification and Control*, 2009. 30(2):71–86. doi:10.4173/mic.2009.2.3.
- Garca, C. E., Prett, D. M., and Morari, M. Model predictive control: Theory and practice A survey. *Automatica*, 1989. 25(3):335–348. doi:10.1016/0005-1098(89)90002-2.
- Gatzke, E. P., Meadows, E. S., Wang, C., and Doyle, F. J. I. Model based control of a four-tank system. *Computers and Chemical Engineering*, 2000. 24(2):1503–1509. doi:10.1016/S0098-1354(00)00555-X.
- Huang, B. and Kadali, R. *Dynamic Modeling, Predictive Control and Performance Monitoring*. Springer, 2008.
- Jazwinski, A. H. *Stochastic processes and filtering theory*. Academic Press, 1989.
- Johansson, K. H. Interaction bounds in multivariable control systems. *Automatica*, 2000. 38(2):1045–1051. doi:10.1016/S0005-1098(01)00285-0.
- Maciejowski, J. *Predictive control: with constraints*. Pearson Education, 2002.
- Maeder, U., Borrelli, F., and Morari, M. Linear offset-free Model Predictive Control. *Automatica*, 2009. 45(2):2214–2222. doi:10.1016/j.automatica.2009.06.005.
- Mayne, D. Q., Rawlings, J. B., Rao, C. V., and Sokaert, P. O. M. Constrained model predictive control: Stability and optimality. *Automatica*, 2000. 36(2):789–814. doi:10.1016/S0005-1098(99)00214-9.
- Pontryagin, L. S., Boltyanskii, V. G., Gamkrelidze, R. V., and Mishchenko, E. F. *The Mathematical Theory of Optimal Processes*. English translation: Interscience 1962., 1956.
- Propoi, A. I. Use of linear programming methods for synthesizing sampled-data automatic systems. *Automation and remote control*, 1963. 24(7):837–844.

- Qin, S. J. and Badgwell, T. A. A survey of industrial model predictive control technology. *Control Engineering Practice*, 2003. 11(11):733–764. doi:[10.1016/S0967-0661\(02\)00186-7](https://doi.org/10.1016/S0967-0661(02)00186-7).
- Ruscio, D. D. On Tuning PI Controllers for Integrating Plus Time Delay Systems. *Modeling, Identification and Control*, 2010. 31(4):145–164. doi:[10.4173/mic.2010.4.3](https://doi.org/10.4173/mic.2010.4.3).
- Ruscio, D. D. Discrete LQ optimal control with integral action: A simple controller on incremental form for MIMO systems. *Modeling, Identification and Control*, 2012. 33(2):35–44. doi:[10.4173/mic.2012.2.1](https://doi.org/10.4173/mic.2012.2.1).
- Ruscio, D. D. and Foss, B. On state space model based predictive control. In *IFAC Symposium on Dynamics and Control of process Systems (DYCOPS-5)*, Corfu, Greece, June. 1998 .
- Widnall, W. S. The Minimum-time Thrust-Vector Control Law in the Apollo Lunar-Module Autopilot. *Automatica*, 1970. 6(6):661–671. doi:[10.1016/0005-1098\(70\)90039-7](https://doi.org/10.1016/0005-1098(70)90039-7).