Stability issues for First Order Predictive Functional Controllers: Extension to Handle Higher Order Internal Models

Mohamed tarek Khadir
Department of Computer Science
University Badji Mokhtar
Annaba, Algeria
Email: Khadir@lri-annaba.net

John Ringwood
Department of Electronic Engineering
National University of Ireland
Maynooth, Ireland
Email: john.ringwood@eeng.may.ie

***Abstract*— Predictive Functional Control (PFC), belonging to the family of predictive control techniques, has been demonstrated as a powerful algorithm for controlling process plants. The first order input/output PFC formulation has been a particularly attractive paradigm for industrial processes, with a combination of simplicity and effectiveness. Though its use of a lag plus delay ARX/ARMAX model is justified in many applications, it may lead to chattering and/or instability. In this paper, instability of first order PFC is addressed,and solutions to handle higher order and difficult systems are proposed.**

Keywords: Model predictive control, predictive functional control, non-minimum phase systems, oscillatory systems.

I. Introduction

Model predictive control grew rapidly in popularity and its field of application diversified substantially since its first applications in the refining and petrochemical industry in 1980 [1], [2]. It is reported in [3], that MPC has been used in over 2,000 industrial applications in the chemical, pulp and paper and food processing industries, on top of the traditional ref ining and petrochemical sector.

Although the principles of MPC are universal, and can be found in many textbooks [4], [5], [6], a wide range of MPC algorithm was developed, primarily to suit given types of industrial application. Among the most popular MPC algorithms one can cite:

- Model Predictive Heuristic Control (MPHC), with the original algorithm called IDCOM for identification and control, and HIECON for hierarchical control most suited for large multivariable systems [7].
- Dynamic Matrix Control (DMC) from Cutler and Ramaker [2].
- Generalised Predictive Control (GPC), [8].
- Predictive Functional Control (PFC), developed by Richalet and ADERSA [9], [4].

For single-input/single-output (SISO) systems, a transfer function internal model formulation, as used in GPC and PFC, is more convenient to manipulate than the over-parameterised step response model used in DMC that requires a large number of parameters, often truncated for a more efficient computation time. Moreover, input/output representations, *e.g.,* ARX/ARMAX are preferred to state space formulations for SISO systems with small turnovers as it does not include the notion of state and matrix calculus. This matches the wish of many industries of a transparent and/or well understood design like PID.

PFC can use many forms of internal model, including state space [10], input/output [9], Finite Impulse Response (FIR) [7], fuzzy rules [11], etc. However, the main distinguishing feature of PFC over other MPC algorithms is that the internal models used are independent internal models, which depend solely on the process input. Industrial vendors ADERSA claim that input output internal models with mixed outputs from the process and the model (or state space models with an estimator) realign the model state on noisy data (output measurements), hence often giving poor predictions and often leads to an offset [12]. The second distinguishing feature of PFC is the construction of the manipulated variable on a set of basis functions [9] *e.g.*, a step input, in the simplest case.

In this paper, the goal is to extend the applicability of the intuitively attractive input/output PFC formulation to a wider range of processes than currently documented in the literature. While first-order (with delay) process models are widely and successfully used over a range of application areas, this paper will attempt to provide higher-order control solutions while retaining the attractive simplicity of the first-order solution. In fact, a number of our extensions rely on the core first-order solution. The paper proceeds (in Section II) by decomposing general SISO ARX and ARMAX models (with real poles) into sets of first order subsystems, using both parallel and cascade forms. Composite PFC solutions for these decomposed systems are developed. Two simulation studies are used to illustrate the effectiveness of the developed control solutions and conclusions are drawn in Section 5.

II. PFC control design

PFC operates on the following four principles [4]:

- Internal model,
- reference trajectory,

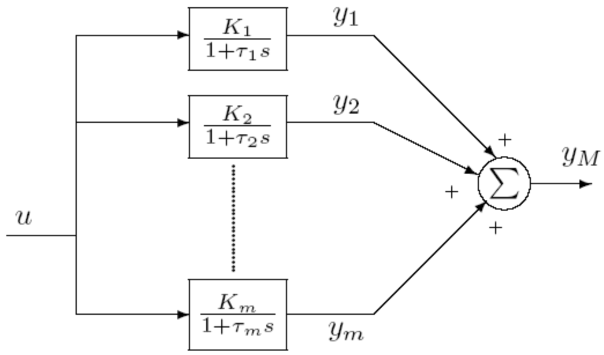


Fig. 1. m^{th} order parallel model

- auto-compensation, and
- calculation of the manipulated variable.

In the case of a higher order process, the internal model needs (ideally) to be of the same order as the process if plant/model mismatch is to be avoided. Observing the fact that any system of order m can be decomposed into a *set* of first order blocks may allow a composite controller to be developed, based on a set of first order PFC controllers. Subsection II-A and document one possible approaches which utilise such a philosophy.

A. Internal model in parallel form

For a high order strictly proper internal process, $G_M(\mathrm{s})$ (1), the transfer function representation based on a parallel decomposition is given in equation (1).

$$G_M(\mathrm{s}) = \sum_{i=1}^{m} \frac{K_i}{1+\tau_i \mathrm{s}} \tag{1}$$

1) Output prediction: From Fig. 1, the model output $y_M(k)$ is given by (2).

$$y_M(k) = y_1(k) + y_2(k) + \cdots + y_m(k) \tag{2}$$

The difference equation obtained from a Zero Order Hold (ZOH) equivalent of the model in (2) is given by:

$$y_i(k) = \alpha_i y_i(k-1) + K_i(1-\alpha_i)u(k-1) \tag{3}$$

$$1 \leq i \leq m$$

where

$$\alpha_i = e^{(-\frac{T_s}{\tau_i})}. \tag{4}$$

with T_s as the sampling period. Replacing equation (3) in (2) gives the model output equation (5):

$$y_M(k) = \alpha_1 y_1(k-1) + \alpha_2 y_2(k-1) + \cdots + \alpha_m y_m(k-1) +$$
$$[K_1(1-\alpha_1) + K_2(1-\alpha_2) + \cdots + K_m(1-\alpha_m)]u(k-1) \tag{5}$$

or, more compactly, as in equation (6):

$$y_M(k) = \sum_{i=1}^{m} \alpha_i y_i(k) + \sum_{i=1}^{m} K_i(1-\alpha_i)u(k-1) \tag{6}$$

where:

$$y_A(k+H) = \sum_{i=1}^{m} \alpha_i^H y_i(k) \tag{7}$$

and

$$y_F(k+H) = \sum_{i=1}^{m} K_i(1-\alpha_i^H)u(k) \tag{8}$$

2) Reference trajectory formulation: The future process output is specified by the reference trajectory, initialised on the real process output, y_P. The reference trajectory used in PFC is generally an exponential given by:

$$y_R(k+H) = C(k) - \lambda^H(C(k) - y_P(k)) \tag{9}$$

where λ is given in equation (10) as:

$$\lambda = e^{\left(-\frac{T_s}{T_R}\right)} \tag{10}$$

with T_R being the desired Closed Loop Response Time (CLRT). At the coincidence horizon h the estimated process output, $\widehat{y}_P$, is set equal to the reference trajectory.

$$y_R(k+H) = \widehat{y}_P(k+H) \tag{11}$$

where the process output estimate $\widehat{y}_P$ is given by:

$$\widehat{y}_P(k+H) = y_M(k+H) + (y_P(k) - y_M(k)) \tag{12}$$

Replacing $y_M(k+H)$ with the expression from equation (6), with $k = k+H$, we obtain:

$$\widehat{y}_P(k+H) = \sum_{i=1}^{m} y_i(k+H) + (y_P(k) - \sum_{i=1}^{m} y_i(k)) \tag{13}$$

3) Computation of the control law: At the coincidence point, $y_R(k+H) = \widehat{y}_P(k+H)$, and using equations (7), (8) and (13) we obtain:

$$C(k)(1-\lambda^H) - y_P(k)(1-\lambda^H) + y_1(k)(1-\alpha_1^H) + y_2(k)(1-\alpha_2^H)$$

$$+ \cdots + y_m(k)(1-\alpha_m^H) = (K_1(1-\alpha_1^H) + K_2(1-\alpha_2^H)$$
$$+ \cdots + K_m(1-\alpha_m^H))u(k) \tag{14}$$

Rewriting the expression in equation (14), we end up with the control law given in (15) as:

$$u(k) = \frac{(C(k) - y_P(k))(1-\lambda^H)}{\sum_{i=1}^{m} K_i(1-\alpha_i^H)} + \frac{\sum_{i=1}^{m} y_i(k)(1-\alpha_i^H)}{\sum_{i=1}^{m} K_i(1-\alpha_i^H)} \tag{15}$$

4) Handling of added disturbances: For the ARMAX case (inclusion of a output disturbance v), a decomposition of the same form as above can be specified as:

$$y(\mathrm{s}) = \sum_{i=1}^{m} \frac{K_i}{1+\tau_i \mathrm{s}} u(\mathrm{s}) + \sum_{i=1}^{m} \frac{K_i'}{1+\tau_i \mathrm{s}} v(\mathrm{s}) \tag{16}$$

Following the steps of Sections II-A.1 to II-A.3, the corresponding PFC control law is given as:

$$u(k) = \frac{(C(k) - y_P(k))(1-\lambda^H)}{\sum_{i=1}^{m} K_i(1-\alpha_i^H)} + \frac{\sum_{i=1}^{m} y_i(k)(1-\alpha_i^H)}{\sum_{i=1}^{m} K_i(1-\alpha_i^H)}$$

$$-\frac{\sum_{i=1}^{m} K_i'(1-\alpha_i^H)}{\sum_{i=1}^{m} K_i(1-\alpha_i^H)} v(\mathrm{s}) \tag{17}$$

Specification of the disturbance dynamics in a parallel from is not crucial to the determination of the controller solution, as long as disturbance is subtracted in a feedforward manner (as in equation (17)). However, the choice as in (16) leads to a particularly elegant control solution.

5) Proper systems: It is also possible to come across proper systems where the order of the numerator and denominator are equal. In this case we can further decompose the system into a gain plus a sum of first order gain/pole systems as shown in equation (18):

$$G_M(\mathrm{s}) = \frac{y_M(\mathrm{s})}{u(\mathrm{s})} = K_0 + \sum_{i=1}^{m} \frac{K_i}{1+\tau_i \mathrm{s}} \tag{18}$$

The difference equations based on the ZOH equivalent is given by:

$$y_0(k) = K_0 u(k) \tag{19}$$

$$y_1(k) = \sum_{i=1}^{m} \alpha_i y_i(k-1) + \sum_{i=1}^{m} \left(K_i(1-\alpha_i)\right) u(k-1) \tag{20}$$

$$y_M(k) = y_0(k) + y_1(k) \tag{21}$$

Developing a PFC control law for such a system gives the following analytical solution:

$$u(k+1) = \frac{(C(k) - y_P(k))(1-\lambda^H)}{K_0}$$

$$+\frac{\sum_{i=1}^{m} y_i(k)(1-\alpha_i) + y_0(k)}{K_0} + \frac{\sum_{i=1}^{m} K_i(1-\alpha_i)}{K_0} u(k) \tag{22}$$

The control law (22) may lead to unstable or ringing MVs, depending on the values of the dominant zero(s) of the system. This is investigated fully in Section III.

B. Tuning, constraint handling and time delay compensation in PFC

As the primary goal of this paper is to develop a generic higher order PFC controller based on an ARX/ARMAX representation, tuning techniques, constraint handling and time delay compensation approaches are not dealt with in detail, since no significant modification from the original form [9], [4] is carried out, at the exception of the work of Rossiter [14] investigating unstable systems. A brief idea about how PFC deals with these issues is given in what follows:

1) Tuning: Often, an exponential reference trajectory is chosen along with a zero order basis function (a step function) [9]. A default choice of $h = 1$ for the co-incidence is appropriate for first order or well behaved systems, while a larger value can be chosen for more emphasis on a smooth MV, which is a common requirement in many industrial systems. However, $h = 1$ is unsuitable for non-minimum phase or oscillatory processes as it may lead to instability, so a co-incidence point beyond the in¤ection points of the transient response should be chosen. Such choices of tuning parameters (reference trajectory, basis function and co-incidence point) result in particularly straightforward control calculations, which are attractive from an intuitive viewpoint. In PFC, the desired response is normally specified as:

$$R_r = \frac{OLRT}{CLRT} \tag{23}$$

defining the ratio of the Open Loop Response Time (OLRT, time to 90% of the final value) to the Closed Loop Response Time (CLRT), T_r, defined in (10). For slow processes, *e.g.,* heat exchange systems, a ratio of 3 is found most suitable [15]. Tuning becomes a one degree of freedom operation, in the selection of T_r, which can be tuned much the same as a gain in PID design [10].

2) Constraint handling: PFC uses a simple (but non-optimal) solution to handle constraints. For input constraints, the model is simply given the constrained input value, rather then the manipulated variable calculated by the PFC algorithm [9]. However, for open loop unstable systems with a factor of the form $\frac{\mathrm{s}-a}{\mathrm{s}-ra}$, $r > 1$, the original constraint handling scheme may lead to instability. Rossiter [14] proposed a modifcation to the original approach to ensure stability when controlling such systems, keeping the simple features of the PFC algorithm. Constraints on the Controlled Variable (CV), is handled using a multiple controller technique, where a separate controller calculates a MV based on a set-point on the actual CV constraints. This MV is used only if the online controller leads the a CV outside the constraints boundaries [4].

3) Time delay compensation: In the linear case, the delay in a system can be referred to the system output. In that case, the delayed value $y_P(k)$ is available but not $y_P(k+d)$.

If we have prior knowledge of the delay value d, then $y_P(k+d)$ can be estimated as:

$$\hat{y}_P(k+d) = y_P(k) + y_M(k) - y_M(k-d) \tag{24}$$

The control laws obtained in equations (15), (17) and (22), are then still valid replacing y_P by its estimate from equation (24)

C. Example 1: Interleaved system

Considering the third order interleaved system:

$$G(s) = \frac{(1+5s)(1+s)}{(1+10s)(1+2s)(1+0.5s)} \tag{25}$$

A simplified model can be obtained using a balanced realisation transformation followed by an order reduction [16] stage. The simplest reduced model can then be obtained in the form of a first order system, and is given by:

$$G'(s) = \frac{0.99}{(1+8s)} \tag{26}$$

If the sampling period is chosen to be $T_s = 0.1$, the control results given by PFC controllers, using the full and the simplif ied systems as internal models, is given in figure 2.

In this case the improvement of using a full internal model over a simplified one is clearly highlighted in figure 2, despite the good results obtained with the latter. Tuned to give roughly the same CLRT, the PFC using a full model gives a much faster control response. The first order PFC can only achieve such speed at the expenses of a much more aggressive MV which may go out of constraints.

III. Appearance of undesirable controller poles

In the case of a proper system, *i.e.,* where the numerator and the denominator are of the same order, the control law must be modified. A controller pole appears, depending on the values of the process zeros (equation (22)). For simplicity, let us consider a PFC development for a first order proper system of the form:

$$G_M(\mathrm{s}) = \frac{K(1+a\mathrm{s})}{1+\tau\mathrm{s}} = \frac{y_M(s)}{u(s)} \tag{27}$$

Following the same development as in Sections II-A.1 to II-A.3, the following control law can be easily obtained:

$$u(k+1) = \frac{\tau(C(k)-y_P(k))(1-\lambda^H)}{Ka} + \frac{\tau y_M(k)(1-\alpha^H)}{Ka} - \left(\frac{\tau}{a}(1-\alpha^H)-1\right)u(k) \tag{28}$$

A. Stability analysis

Observe that the control law in equation (28) can be represented in the z-domain by:

$$u(\mathrm{z}) = \frac{N(z)}{\mathrm{z}-1+\frac{\tau}{a}(1-\alpha^H)} \tag{29}$$

where $N(z)$ depends on the particular internal model formulation. It can be seen that the controller contains a pole given by:

$$\mathrm{z} = 1 - \frac{\tau}{a}(1-\alpha^H) \tag{30}$$

If:

1) $a > \tau(1-\alpha^H)$ then $0 < \mathrm{z} < 1$ which gives a stable manipulated variable with no ringing.
2) $a < \tau(1-\alpha^H)$ then $\mathrm{z} < 0$ and $u(k)$ will oscillate with period $2T_s$.
3) $a < \frac{\tau}{2}(1-\alpha^H)$, including $a < 0$, then $\mathrm{z} < -1$ thus the controller is unstable.

Clearly, an unstable or oscillatory manipulated variable is undesirable and some modification of the PFC algorithm in (28) is required. One possible solution, is to decompose the system in (27) as:

$$G_M(\mathrm{s}) = K_0 + \frac{K_1}{1+\tau\mathrm{s}} = \frac{K(1+a\mathrm{s})}{1+\tau\mathrm{s}} \tag{31}$$

where:

$$K_0 = \frac{Ka}{\tau} \quad \text{and} \quad K_1 = K - \frac{Ka}{\tau}. \tag{32}$$

Since neither individual system contains a zero, we can utilise the control solution for parallel subsystems, as in (15). However, it is found that such a formulation still results in a controller pole, as in equation (30), appearing *implicitly* in the overall control calculation.

Nevertheless, such an approach can lead to an improvement, if a minor adjustment in the process model is allowed. Consider the approximation to the ZOH equivalent of (31) as:

$$y_0(k) = K_0 u(k-1) \tag{33}$$

$$y_1(k) = \alpha y_1(k-1) + K_1(1-\alpha)u(k-1) \tag{34}$$

$$y_M(k) = y_0(k) + y_1(k) \tag{35}$$

with an introduction of a 1-step (extra) delay into the pure gain term in equation (33). The control development then consists of controlling a sum of two systems:

- a gain/delay system $y_0(\mathrm{s}) = K_0 e^{T_s s} u(\mathrm{s})$, and
- a gain/pole system, $y_1(\mathrm{s}) = \frac{K_1}{1+\tau\mathrm{s}} u(\mathrm{s})$.

where the composite prediction model autoregressive and forced responses are given by:

$$y_A(k+H) = \alpha^H y_1(k) \tag{36}$$

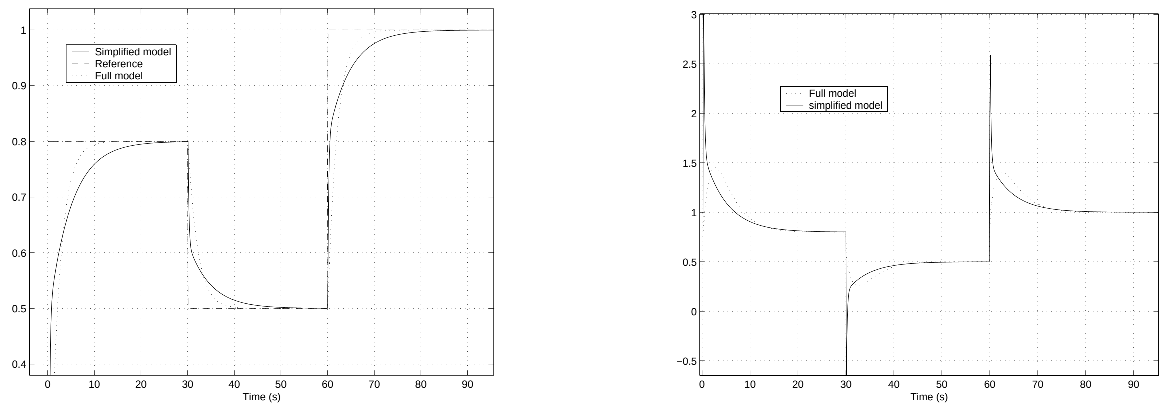
$$y_F(k+H) = (K_0 + K_1(1-\alpha^H))u(k) \tag{37}$$

Following the previous development, as in Sections II-A.1 to II-A.3, the final control law is given by:

$$u(k) = \frac{(C(k)-y_P(k))(1-\lambda^H)}{K_1(1-\alpha^H)+K_0} + \frac{y_1(k)(1-\alpha^H)+y_0(k)}{K_1(1-\alpha^H)+K_0} \tag{38}$$

Equation (38), can be recast to show the controller pole by explicitly writing y_0 and y_1 in terms of $u(k-1)$ to give:

$$u(k) = \frac{(C(k)-y_P(k))(1-\lambda^H)}{K_1(1-\alpha^H)+K_0} + \frac{\alpha y_1(k-1)(1-\alpha^H)}{K_1(1-\alpha^H)+K_0}$$



(a) Controlled variable

(b) Manipulated variable

Fig. 2. PFC response for a 3rd order interleaved system

$$+\frac{K_1(1-\alpha^H)(1-\alpha)+K_0}{K_1(1-\alpha^H)+K_0}u(k-1) \tag{39}$$

with the controller pole identified as:

$$\mathrm{z}=\frac{K_1(1-\alpha^H)(1-\alpha)+K_0}{K_1(1-\alpha^H)+K_0} \tag{40}$$

Using the definitions of α, K_0 and K_1 from (4) and (32),

1) If $a > 0$ then $0 < \mathrm{z} < 1\ \forall\ |a|$, which gives a stable manipulated variable with no ringing.
2) If $a < 0$ (non-minimum phase zero), then $\mathrm{z} > 1$ and the controller is unstable. This case is dealt with in Section III-C.

It is clear that such a formulation shifts the controller pole to the positive real axis, which solves the ringing problem. This is illustrated by Example 2, Section III-B.

Another possible way to eliminate the difficulties caused by the introduction of a controller pole is to perform a factorization of the process zero polynomial, as is common in other control formulations, such as pole placement [17]. In this philosophy, zeros which cause controller instability or ringing are separated from the plant zero polynomial and can, if desired, be put into the reference model (as is done in [17]). In our case, such zeros are simply discarded (with preservation of the dc gain). However, although PFC has been shown to be relatively robust to plant/model mismatch [15], it was noted in [**?**] that this mismatch may become significant as a gets larger, possibly affecting controller accuracy. Therefore, this model simplification is very much restricted to well behaved processes and will not be investigated further in this paper.

B. Example 2: Pole/zero system

Given the system:

$$y(\mathrm{s})=\frac{K(1+a\mathrm{s})e^{-d\mathrm{s}}}{1+\tau\mathrm{s}}u(s)+\frac{K_2}{1+\tau\mathrm{s}}v(s) \tag{41}$$

with the parameter values given in Table I.

Parameter	K	a	K_2	τ	d
	1	0.5	2	30	10

TABLE I

EXAMPLE 2 PARAMETER VALUES

An exact, but delay-free, internal model is given by:

$$y_{M1}(\mathrm{s})=\frac{K(1+a\mathrm{s})}{1+\tau\mathrm{s}}u(s)+\frac{K_2}{1+\tau\mathrm{s}}v(s) \tag{42}$$

Further decomposing the pole/zero system, the internal model can be approximated as:

$$y_{M2}(\mathrm{s})=\left(\frac{Ka}{\tau}e^{-T_s s}+\frac{K-\frac{Ka}{\tau}}{1+\tau\mathrm{s}}\right)u(s)+\frac{K_2}{1+\tau\mathrm{s}}v(s) \tag{43}$$

The control performance of PFC controllers, based on the two different internal models (M1 and M2, as given in (42) and (43) respectively), and including time delay compensation as per (24), are given in Figure 3 for $a = 0.5$.

It can be seen from Figure 3 that, although both controllers give good control, the manipulated variable given by the controller base on M1 sustains heavy ringing. This is caused by the the presence of a controller pole between 0 and -1 (see Section III-A).

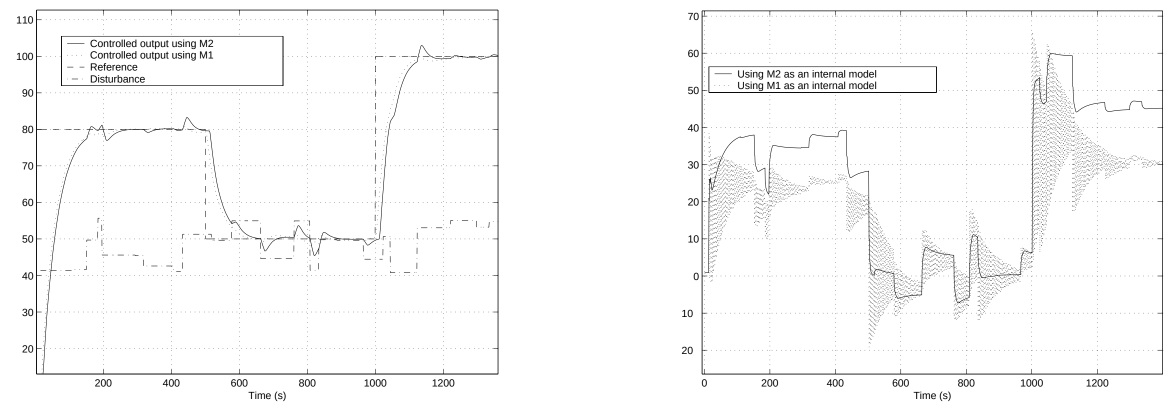
C. Extension to the higher order case

A higher order proper system, in a parallel form can be given by:

$$G_M(\mathrm{s})=K_0+\sum_{i=1}^{m}\frac{K_i}{1+\tau_i\mathrm{s}} \tag{44}$$

If desired, the internal model delay modification, as in (33) may be made to (19) to avoid potential ringing on the MV, with the modified controller calculation of:

$$u(k)=\frac{(C(k)-y_P(k))(1-\lambda^H)}{\sum_{i=1}^{m}K_i(1-\alpha_i)+K_0}+\frac{\sum_{i=1}^{m}y_i(k)(1-\alpha_i)+y_0(k)}{\sum_{i=1}^{m}K_i(1-\alpha_i)+K_0} \tag{45}$$



(a) Controlled variable

(b) Manipulated variable

Fig. 3. PFC performance using y_{M1} and y_{M2} as internal models (with a=0.5)

Note that any input disturbance can always be handled as in equation (17).

IV. Conclusion

This paper has developed higher-order solutions to SISO processes, based on ARX/ARMAX input/output process descriptions, retaining the intuitive appeal of such PFC formulations. Combining the decomposition techniques of Section II with the further developments in Sections 3 and 4, most SISO industrial processes can be handled, with one, or at most 2 parameters, to tune *i.e.,* T_r and h. The choice of h is process and MV/CV trade off dependant, but still results in a choice of a single parameter. This makes PFC easier to tune than (for example) PID, qualifying it as an ideal candidate for industrial use where good dynamic performance and intuitive appeal is paramount.

However, such a compact form is not possible with oscillatory or non-minimum phase systems, since the control parameters have to be calculated from the system's free response utilising a form of the unfactored process difference equation. The reader is directed to [18] for a suitable PFC formulation handelling such systems.

Though extra computational expense is incurred in higher-order controllers, this is not problematic in these days of cheap computational power and the performance advantage is demonstrated clearly in our illustrative examples. Most importantly, the computation is minimised by utilising an input/output formulation (since matrix computations, often containing zero elements, are avoided) and the simplicity and intuitive appeal are maximised.

References

[1] RICHALET, J., RAUL, A., TESTUD, J.L., and PAPON, J.: Algorithmic control of industrial processes. *Proceedings of the* 4^{th} *IFAC Symposium on Identification and System Parameters Estimation*, 1976, pp. 1119-1167.

[2] CUTLER, C.R., and Ramaker, P.S.: Dynamic matrix control - a computer algorithm, *Proceeding of the Joint Automatic Control Conference*, 1980.

[3] QIN, S.J., and BADGWELL, T.A.: An overview of industrial model predictive control technology. *Chemical Process Control. AIChe Symposium Series. Kantor., J., Garcia, C. and Carnahan, B. Eds. New York,*1997, pp. 232-256.

[4] RICHALET, J.: *Pratique de la commande predictive*. Paris: Traite des Nouvelles Technologies, Serie Automatique Hermes, 1993.

[5] MORARI, M., and LEE, J.H.: Model predictive control: past present and future. *Comp. Chem. Engineering,* 2000, 23, pp. 667-682.

[6] MACIEJOWSKI, J.M.: *Predictive control with constraints.* England: Pearson Education Limited, 2002.

[7] RICHALET, J., RAUL, A., TESTUD, J.L., and PAPON, J.: Model predictive heuristic control: Application to industrial processes. *Automatica,* 1978, 14, pp413-428.

[8] CLARKE, D.W., MOHTADI, C., and TUFFS, P.S.: Generalised predictive control-part I. The basic algorithm. *Automatica,* 1987, 23(2), pp. 137-148.

[9] RICHALET, J.: La commande predictive, *Techniques de l'Ingenieur Traite Mesure et Control,* 1998, R7 423, pp. 1-17.

[10] ROSSITER, J.A., and RICHALET, J.: Predictive functional control of unstable processes. *Sheffeld University, Dept. of Automatic Control and Systems Engineering, internal report 807*, 2001.

[11] SKRJANK, I., and MATKO, D.:*Predictive functional control based on fuzzy model for heat-exchanger pilot plant* . IEEE Transactions on Fuzzy Systems, 8(6), 2000, pp.705 -712.

[12] ROSSITER, J.A., and RICHALET, J.: Realigned models for prediction in MPC: a good thing or not?. *Proceeding of the APC 6*, 2001.

[13] HOUPIS, C.H, and LAMONT, G.B.: *Digital control systems*. McGraw-Hill International Edition, 1992.

[14] ROSSITER, J.A., and RICHALET, J.: Handling constraints with predictive functional control of unstable processes. *Proceeding of American Control Conference*, 2002.

[15] KHADIR, M.T.: *Modelling and predictive control of a milk pasteurisation plant*. Ph.D Thesis. Dept. of Electronic Engineering, NUI Maynooth, Ireland, 2002.

[16] MOORE, B.C.: Principal component analysis in linear systems: controlability, observability, and model reduction. *IEEE on Automatic Control,* 1981, 26(1), pp. 17-32.

[17] ASTROM, K.J. and WITTENMARK, B.: *Computer-controlled systems: theory and design (Third Ed.)*, Prentice-Hall, 1997.

[18] KHADIR, M.T.: *The simplest Model Predictive Controller for Non-Minimum Phase behavior and Oscillatory Systems*. Proc. Of the 6th Portuguese Conference on Automatic Control. Faro, Protugal, Vol1, pp 255-260, 2004.