Control and Intelligent Systems, Vol. 31, No. 1, 2003

LINEAR AND NONLINEAR MODEL PREDICTIVE CONTROL DESIGN FOR A MILK PASTEURIZATION PLANT

M.T. Khadir* and J.V. Ringwood*

Abstract

This article investigates the design of linear and nonlinear model predictive controllers (MPCs) in order to improve the control of pasteurization temperature in a milk plant. MPC schemes required the development of a prediction model for use internally within the controller. An artificial neural network (ANN) model of the plant is established and validated. A linearized model is then obtained around the operating point from the ANN model. The linearized and the ANN models are used for prediction for the linear and nonlinear predictive controllers, respectively. The MPC responses are compared with a benchmark PID controller behaviour, the parameters of which have been tuned to minimize the same criteria as used for the predictive controllers.

Key Words

Milk pasteurization, nonlinear modelling, model predictive control, artificial neural networks

1. Introduction

Predictive control is becoming a valuable control strategy for higher control requirements, such as tighter, faster regulation or tracking in the industrial world. Model predictive controllers (MPCs) have been used in over 2,000 industrial applications in the refining, petrochemical, chemical, pulp and paper, and food processing industries [1]. Some examples of industrial predictive controllers include PFC from ADERSA [2] and dynamic matrix control (DMC) from DMC Corp [3]. Most of these algorithms rely on linear or linearized internal models [1, 4]. Using predictive control, a process is regulated by specifying the desired plant output at a particular instance or instances in the future and then calculating the controller action that minimizes the predicted error either in the form of an analytical solution, for linear internal models with no constraints, or using an optimizer in the case of a nonlinear internal model.

* School of Electronic Engineering, National University of Ireland, Maynooth, County Kildare, Ireland; e-mail: {tarek.khadir, john.ringwood}@eeng.may.ie

(paper no. 201-1237)

Both linear and nonlinear MPC formulations offer advantages in their own way. A linear formulation has a computationally attractive analytical solution, provided constraint handling is not required. A nonlinear formulation, on the other hand, can deal with operating-point dependencies via a comprehensive nonlinear internal model, and constraint handling can be easily incorporated within the numerical optimization required to solve for the control signals. However, a high computational price is paid in order to perform this numerical optimization at each sampling instant; although with a sampling period of the order of tens of seconds, the pasteurization application easily permits such an investigation.

Despite the recent proliferation of industrial applications of predictive control, and the potential benefits obtainable in a pasteurization context, little interest has been shown in applying MPC to the pasteurization problem [5]. In this article, MPC is applied to a milk pasteurization plant in order to improve the control performance in terms of mean absolute error (MAE), overshoot, and maximum variance in steady state (MVSS). The reason for developing a new control approach is that current control approaches require an artificially high temperature setpoint (e.g., 75.0°C) due to large temperature variance (up to 3.0°C, in this particular case). Indeed, the pasteurization temperature is only 72.0°C [6]; however, because of the variance introduced by the PI or PID controller due to time delay, constraints, etc., a 3.0°C safety margin is taken. If the control is improved and the variance reduced, then a setpoint closer to 72.0°C might be considered. This will allow energy savings and avoid overheating the milk, which may give it a burnt taste and degrade its nutritional value.

2. Milk Pasteurization Process

The pasteurizer used is a Clip 10-RM plate heat exchanger (PHE) from Alfa Laval. A PHE consists of a pack of stainless steel plates clamped in a frame. The plates are corrugated in a pattern designed to increase the flow turbulence of the medium and the product [6]. The pasteurizer is divided in five sections, S1–S5. Sections S4 and S2

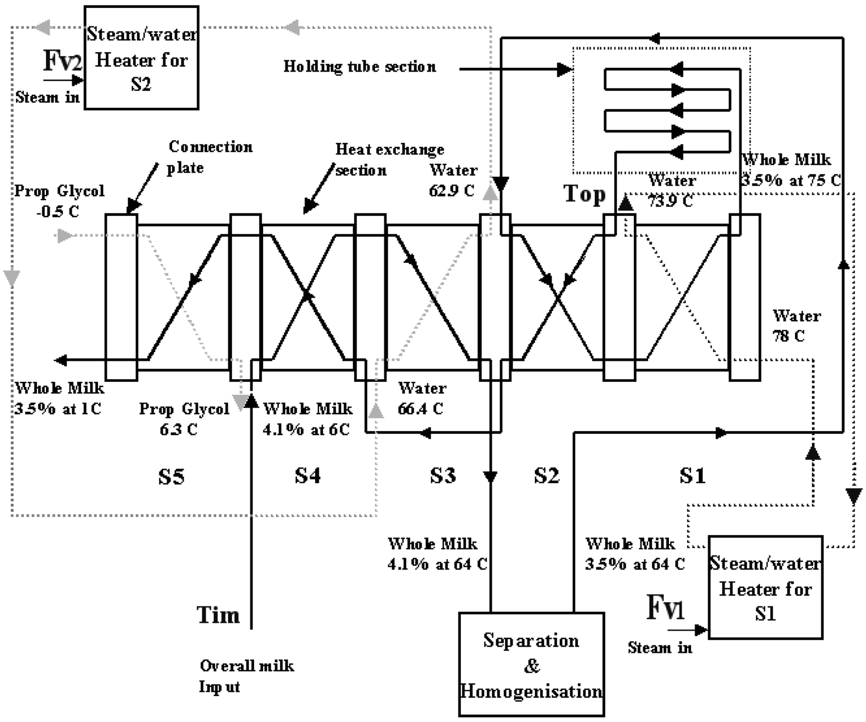


Figure 1. General layout of the pasteurizer.

are for regeneration (i.e., by regeneration we mean heating of the incoming milk by the already heated milk [6]), S1 and S3 for heating, and S5 for cooling. In the Clip 10-RM, the milk treatment is performed as shown in Fig. 1. First, the raw milk at a concentration of 4.1% fat enters section S4 of the PHE at a temperature of 2.0°C. It is then preheated to a temperature of 60.5°C by the outgoing pasteurized milk, which as a result is reduced to a temperature of 11.5°C. Passing this section, the milk, now at a temperature of 60.5°C, enters section S3, where its temperature increases to 64.5°C using hot water as a medium. The milk, before reaching the next section, is first separated from the fat and then standardized and homogenized to a concentration of 3.5%. It then enters section S2, where it is preheated to a temperature of 72.0°C using the already pasteurized milk as a medium. The milk is finally brought to the pasteurization temperature, T_{op}, in section S1 (75.0°C) using hot water at around 77.0°C as a medium. After that, the homogenized pasteurized milk is held at the pasteurization temperature for 15 s in the holding tube section before being cooled using the incoming cold milk in sections S4 and S2. Finally, the pasteurized milk enters the cooling section (section S5) at a temperature of 11.5°C. The milk is chilled to a temperature of 1.0°C using propylene glycol as a medium at a temperature of −0.5°C. Note that the water for the heating sections S3 and S4 is brought to the adequate temperature in steam/water heaters of type CB76 from Alfa Laval. As shown in Fig. 1, milk pasteurization temperature is a function of three inputs: steam flow injected in steam/water heater for S1, steam flow injected in steam/water heater for S2, and the milk input temperature, defined as F_{v1}, F_{v2}, and T_{im}, respectively. The milk pasteurization temperature is then given by a multi-input single-output (MISO) system, having F_{v1}, F_{v2}, and T_{im} as inputs and T_{op}, the milk pasteurization temperature, as output.

3. ANN Modelling of the Pasteurization Plant

The ANN model used in this article has been established and validated in [7]. For ease of training and overall reduction in neuron count, a multilayer network with an input, an output, and two hidden layers is used. The inputs to the ANN model are F_{v1}, F_{v2} one step delayed, and their previous values, and eight delayed values of the process output T_{op} (see Fig. 2). The output prediction will be given by the ANN model, given by function f_{ANN} (see (1)). The final result is obtained after appropriate training and validation:

$$y_{ANN}(k) = f_{ANN}(T_{op}(k-1), T_{op}(k-2), \ldots, T_{op}(k-8), F_{v1}(k-1), F_{v1}(k-2), F_{v2}(k-1), F_{v2}(k-2)) \quad (1)$$

The choice of the inputs has been heavily dictated by the *a priori* information gathered from the first principle physical model developed and used in [8]. As the output pasteurization temperature can be modelled by an eighth-order linear system, this justifies the use of eight delayed signals of $T_{op}(k)$ in (1). The input milk temperature T_{im} is not used as an input in the ANN model formulation, as the milk is kept at a relatively constant temperature around 2.0°C, and its use in the training process will only introduce a random disturbance to be modelled. A topology large enough to permit good modelling and possible network pruning [9] is chosen as 8-12-1. The network was trained for 20,000 epochs using a set of data containing subsets obtained during a series of test protocols on an industrial Clip 10-RM pasteurizer at a sampling rate of 12 s, where

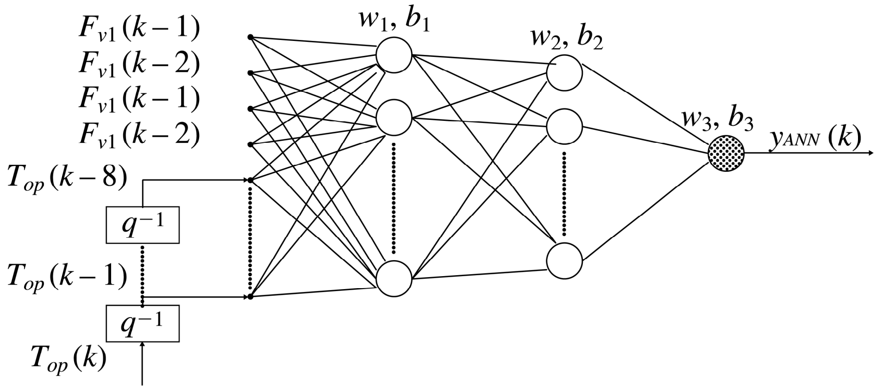


Figure 2. ANN topology and input signals used for training.

F_{v1} and F_{v2} were varied around the operating region of interest. Four subsets of data were used for training, where a separate subset was used for validation in order to obtain an appropriate model. To avoid overtraining (deterioration of the model as it tries to fit the training set [10]), a sum squared error (SSE) on the validation set is plotted and the model parameters are chosen when the SSE is minimum, that is, early stopping [7]. A cross-validation method [10, 11] is used, where all data sets are used for validation in turn and the rest of the data is used for training. This method has proven to be useful when the number of data points is constrained. Moreover, this approach will give a better degree of confidence to the estimates. The definitive ANN model is then given by a linear combination of the models obtained with each validation set in the cross-validation process. The model response is given in Fig. 3(a). The ANN model gives a good approximation of the plant, as the SSE on the normalized (divided by a factor of 100) validation set is equal to (1.4913×10^{-4})°C.

A linear auto-regressive model with exogenous inputs (ARX) model is obtained by linearizing the ANN model at around the operating point of 74–75°C. The simplest linear model structure is obtained, that is, a first-order system with d, the pure delay being equal to one sample period (12 s). The model's form is:

$$y_M = \frac{K_1 \mathrm{e}^{-\mathrm{s}d}}{1+\tau_M \mathrm{s}} F_{v1} + \frac{K_2 \mathrm{e}^{-\mathrm{s}d}}{1+\tau_M \mathrm{s}} F_{v2} \quad (2)$$

where s is the Laplace operator.

The discrete zero-order hold equivalent of the transfer function equation (2) leads to the following difference equation:

$$y_M(k) = \alpha y_M(k-1) + (1-\alpha) K_1 F_{v1}(k-d-1) + (1-\alpha) K_2 F_{v2}(k-d-1) \quad (3)$$

where $\alpha = \mathrm{e}^{(-T_s/\tau_M)}$, with T_s being the sampling period.

The final numerical values for the ARX model, for a sampling time $T_s = 12$ s, is:

$$y_M(k) = 0.9896 y_M(k-1) + 7.6700 F_{v1}(k-2) + 1.2000 F_{v2}(k-2) \quad (4)$$

The first-order model (4) is selected in order to implement a basic PFC (see Section 4). We can see from

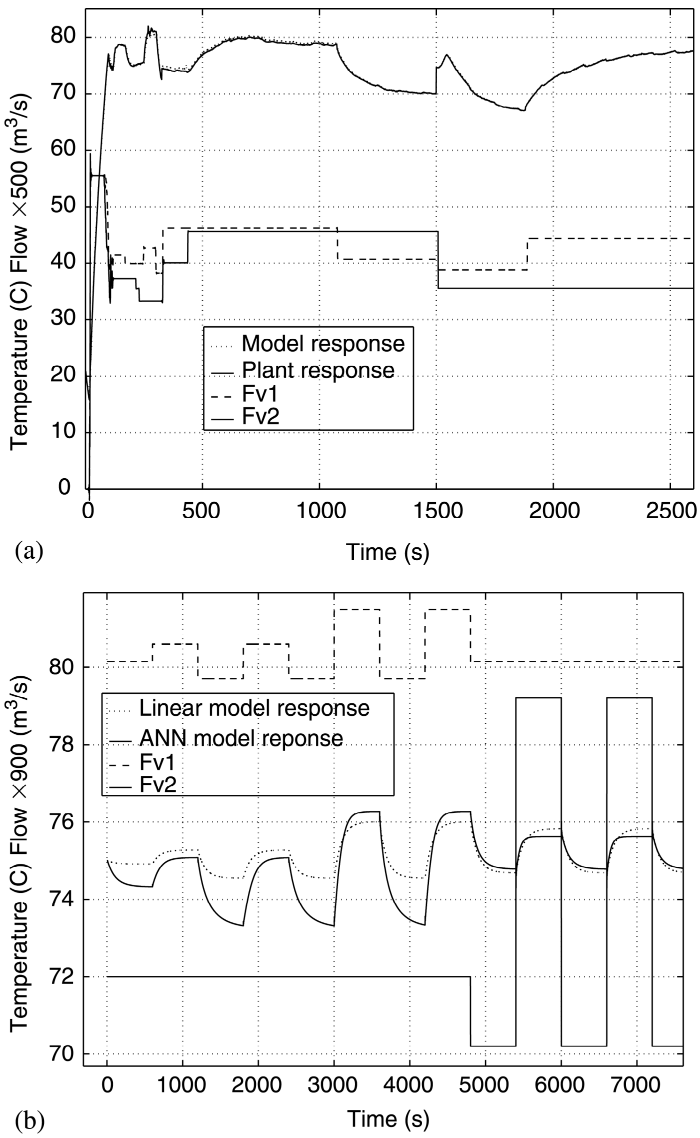


Figure 3. Nonlinear and linearized models of the pasteurizer. (a) ANN model response; (b) linearized model response.

Fig. 3(b) that the linearized model follows the ANN model behaviour around the operating point; however, the mismatch becomes larger further away from that temperature. Such a plant/model (given here by ANN model/linear model) mismatch is a challenge for any predictive controller.

4. Predictive Functional Controller (PFC)

As stated in Section 1, all MPCs use the same basic approach: prediction of the future plant outputs and calculation of the manipulated variable for an optimal control. Most MPC strategies are based on the following principles:

- Use of an internal model
- Specification of a reference trajectory
- Determination of the control law

However, there are many feasible implementations of MPC (see Section 2). The most popular ones for single-input single-output (SISO) systems are GPC [12] and PFC [2]. A broad review of MPC is given in [1, 4]. In this work, PFC has been chosen because of its large number of applications on real industrial processes [1, 2].

PFC was developed by Richalet *et al.* [2]. PFC breaks up the model into an auto-regressive part and a forced part, the latter being a function of the control variable, u. The forced response is projected into a functional basis [2, 13]. In the case where the simplest version of the PFC algorithm is implemented:

- The reference trajectory is an exponential that requires only one initialization point and gives critically damped responses.
- The coincidence horizon is limited to one point $H = 1$.
- The internal model chosen is the linearized first-order ANN model developed in Section 3.
- The manipulated variable is structured as $u(k) \times OB_0(H)$, where OB_0 is a step [2].

4.1 Model formulation

If we consider the first-order linear model obtained in (2), and we define the following linear transformation:

$$u = K_1 F_{v1} + K_2 F_{v2} \tag{5}$$

the goal of the transformation (5) is to obtain a typical first-order transfer function equation (6) in order to implement a basic first-order PFC:

$$y_M = \frac{K_M}{1 + \tau_M \mathrm{s}} u \tag{6}$$

Note that the time delay is not considered in the internal model formulation, and in this case K_M is equal to one. The discrete time formulation of the model zero-order hold equivalent is then obtained as:

$$y_M(k) = \alpha y_M(k-1) + K_M(1-\alpha)u(k-1) \tag{7}$$

where $\alpha = \mathrm{e}^{(-T_s/\tau_M)}$. If the manipulated variable is structured as a step basis function:

$$y_L(k+H) = \alpha^H y_M(k) \tag{8}$$

$$y_F(k+H) = K_M(1-\alpha^H)u(k) \tag{9}$$

where y_L and y_F are, respectively, the free (auto-regressive) and the forced response of y_M.

4.2 Reference Trajectory Formulation

If y_R is the expression of the reference trajectory, then at the coincidence point H:

$$C(n+H) - y_R(k+H) = \lambda^H (C(k) - y_P(k)) \tag{10}$$

Thus:

$$y_R(k+H) = C(k) - \lambda^H (C(k) - y_P(k)) \tag{11}$$

where y_P, the process output, is given in this case by the nonlinear model output y_{ANN}, given in (1), and $C(k)$ is the set point reference.

4.3 Predicted Process Output

The predicted process output is given by the model response, plus a term, given the error between the same model output and the process output:

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

4.4 Computation of the Control Law

At the coincidence point H:

$$y_R(k+H) = \hat{y}_P(k+H) \tag{13}$$

Using (8), (9), (11), and (12), we obtain:

$$C(k) - \lambda^H (C(k) - y_P(k)) - y_P(k) = y_M(k+H) - y_M(k) \tag{14}$$

Replacing $y_M(k+H)$ by its equivalent in (8) and (9), we obtain:

$$\begin{aligned} C(k)(1-\lambda^H) - y_P(k)(1-\lambda^H) + y_M(k)(1-\alpha^H) \\ = K_M(1-\alpha^H)u(k) \end{aligned} \tag{15}$$

Solving for $u(k)$, we obtain the following control law as the final result:

$$u(k) = \frac{(C(k) - y_P(k))(1-\lambda^H)}{K_M(1-\alpha^H)} + \frac{y_M(k)}{K_M} \tag{16}$$

F_{v1} is then obtained, working backward from (5), as u and F_{v2} are known.

4.5 Constraint Handling in PFC

For the input constraints on F_{v1} considered in this article, PFC uses a suboptimal approach to handle those type of constraints. A simple solution will be to feed the model, not with the manipulated variable calculated by the PFC algorithm, but with its constrained value of the same manipulated variable. The model output y_M is then calculated with the new applied manipulated variable [2].

In this paper, a first-order PFC is chosen over other linear MPC schemes for the temperature control in the milk plant, primarily because of its simple internal model. Moreover, PFC has been proven successful for a wide range of industrial processes [1]. Regulation results of a PFC using an internal first-order linear model of the plant, (2), are given in Section 7.

5. Nonlinear Neural Predictive Controller Design

When a nonlinear model is used in an MPC implementation for prediction, an analytical solution cannot be achieved. The use of an optimizer in that case is necessary. In this section, a neural model predictive controller (NMPC) is designed as shown in Fig. 4. The control variable u is obtained by minimizing a criterion function J given in (17):

$$J = \sum_{i=d}^{d+N} [y_R(k+i) - y_{ANN}(k+i|k)]^2 + \sum_{i=0}^{i=N2} \lambda[u(k+i)]^2 \quad (17)$$

where d is the time delay (if any), N the prediction horizon, and $N2$ the control horizon. Note that only output errors after the time delay d are costed, where the prediction model used is the ANN model developed in Section 3. In order to use this model for multistep prediction, we simply replace the known measurements with predicted ones. $\hat{T}_{op}$, a prediction for the process output T_{op}, can then be given as:

$$\hat{T}_{op} = f_{ANN}(\hat{T}_{op}(k-1), \hat{T}_{op}(k-2), \ldots, \hat{T}_{op}(k-8), F_{v1}(k-1), F_{v1}(k-2), F_{v2}(k-1), F_{v2}(k-2)) \quad (18)$$

Where available, real plant measurements are of course used instead of predictions (available for the start of the prediction horizon).

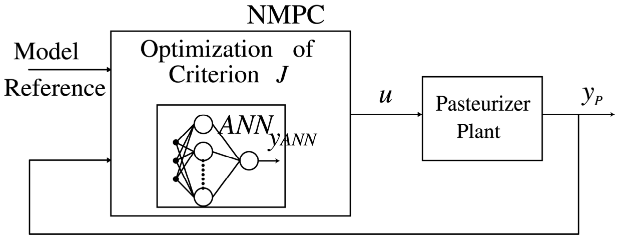


Figure 4. NMPC structure.

A quasi-Newtonian algorithm is used for the function minimization [14]. At each instant k, the predicted output $y_{ANN}(k+i|k)$ is compared to a reference $y_R(k+i)$ describing the optimal trajectory to reach the target $C(k+i)$, subject to input constraints on the manipulated variable. If no reference trajectory is predefined, then $y_R(k+i) = C(k+i)$. The manipulated variable u is represented physically by F_{v1}, where F_{v2} is used to act on the intermediate temperature at the output of section S3 (see Fig. 1) and can be considered in the control of the pasteurization temperature as a disturbance.

6. Optimized Benchmark PID

The PID controller transfer function is usually given by:

$$C(s) = K_p + \frac{K_i}{s} + K_d s \quad (19)$$

where K_p, K_i, and K_d are, respectively, the proportional, integral, and derivative gains. A digital version of the classical PID is used when the plant is operated by any digital/computer-based controller, and can be given by the following set of equations, assuming numerical integration derivative approximations obtained using the backward difference:

$$e_r(k) = y_{ref}(k) - y(k) \quad (20)$$

$$s(k) = s(k-1) + e_r(k) \quad (21)$$

$$u(k) = K_p\left(e_r(k) + \frac{T_s}{T_i}s(k) + \frac{T_d}{T_s}(e_r(k) - e_r(k-1))\right) \quad (22)$$

where $T_i = K_p/K_i$, $T_d = K_d/K_p$, and T_s is the sampling period.

In order to perform a meaningful comparison of MPC and benchmark PID controllers, we strive to use a similar performance criterion for both types of controller. In the MPC case, u is directly determined as the result of an analytical or numerical optimization, whereas in the PID case fixed controller parameters are used. As u is a function of the controller parameters, we can utilize a similar optimization on K_p, T_i, and T_d via the presence of u in the performance criterion, J, (17). In accordance with the linear MPC, a single optimization is performed for the linear PID controller.

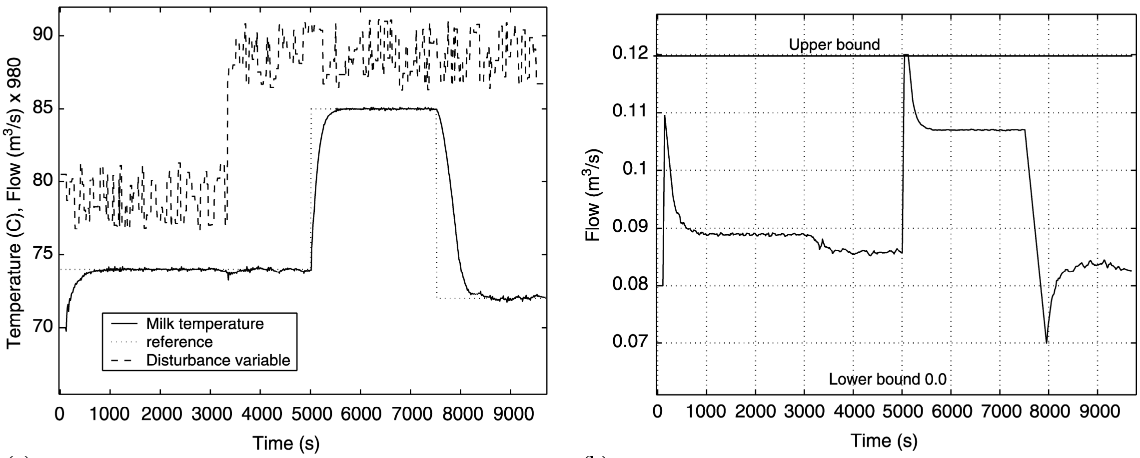
The "optimal" PID parameters are found by performing an offline optimization during a test period time, where:

- the ANN model is used to model the process behaviour;
- the same reference trajectory as in NMPC may be used;
- the quadratic criterion J, (17), may be used as an objective function; and
- the variables returned by the optimization routine are K_p, T_i, and T_d.

Note that the optimization is done offline and therefore is not time restricted. The designer has the freedom to investigate several possibilities of set-point changes and disturbance rejection. In order to be robust, the optimization should investigate large set-point changes as well as input disturbances.

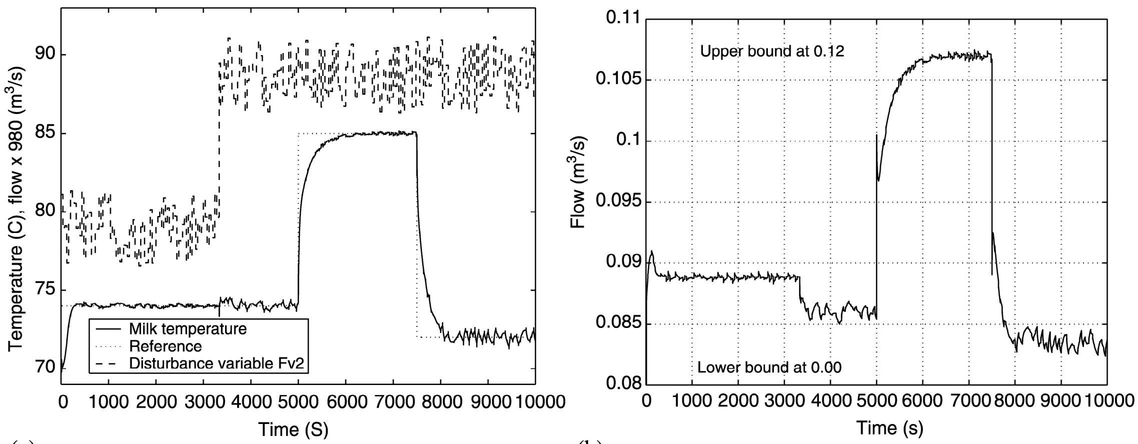
7. Simulation Results

A linear PFC and a nonlinear predictive controller (NMPC) are designed according to Sections 4 and 5, using as internal models the linearized and the ANN model, respectively. For comparison, the benchmark PID described in Section 6 is used. Note that in order to perform a meaningful comparison, all three controllers are subjected to the same modelled disturbance shown along with the controlled variable in Figs. 5(a), 6(a), and 7(a). The prediction horizon



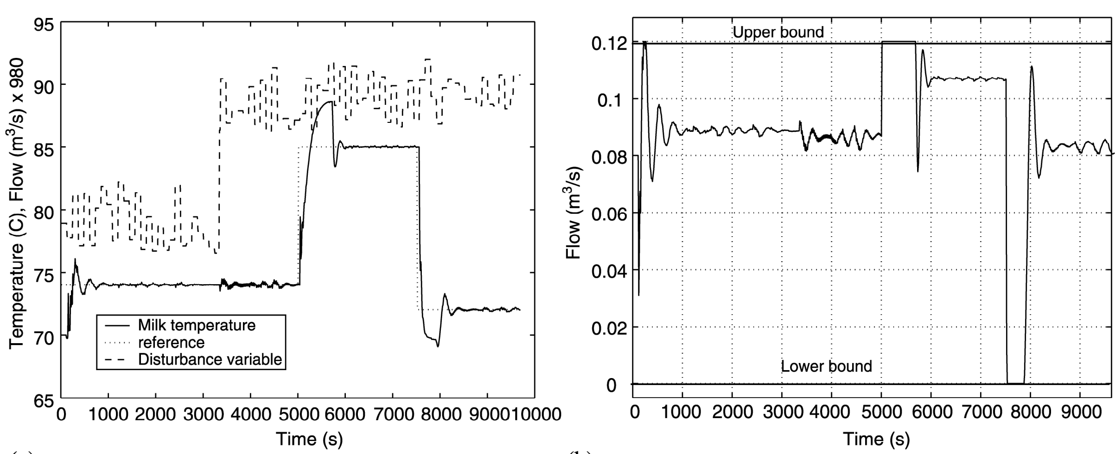
(a) (b)

Figure 5. NMPC control performance. (a) Controlled variable; (b) manipulated variable.



(a) (b)

Figure 6. Linear MPC (PFC) performances. (a) Controlled variable; (b) manipulated variable.



(a) (b)

Figure 7. PID control performances. (a) Controlled variable; (b) manipulated variable.

for NMPC is chosen to be 25 samples (300 s) and the time delay is taken as one sample (close to 15 s holding time in the pasteurizer (Section 2)).

The control performance of the two MPC strategies is shown in Figs. 5(a) and 6(a) for NMPC and PFC, respectively. Fig. 7(a) shows the response given by the optimally tuned PID. The shape of the manipulated variable is given in Figs. 5(b), 6(b), and 7(b) for the NMPC, PFC, and PID controllers, respectively.

Table 1
Performance of the Three Controllers

Control Strategy (°C)	PID	PFC	NMPC
MAE	1.53	0.85	0.36
Maximum overshoot	3.10	0.20	0.00
MVSS	1.10	0.84	0.40

The performance of the three controllers is given in terms of MAE, overshoot, and maximum temperature variance in steady-state (MVSS) in Table 1. We can clearly see from Figs. 5(a), 6(a), and 7(a) and Table 1 that MPC controllers perform better than the PID. PFC permits an analytical solution that, from a computation point of view, is the cheapest option, and even though constraints are not dealt with in an optimal way [2], PFC still gives excellent constraint handling, as shown in Fig. 6(b). The performance of the PFC is similar to that of the NMPC around the set point (74–75°C), where the linearized model best fits the ANN model. At higher or lower reference temperatures, PFC sustains a slight increase in variation due to the increasing mismatch between the plant model (nonlinear ANN) and the internal first-order linear model, but still performs better than PID and is found to exhibit good robustness. PFC performance can be improved by using a higher-order linearized model, but the improvements are not found to be worthwhile. NMPC is found to have the best control capability over the full range of the validated ANN model. This performance is obtained at the expense of using an online optimization and a complicated nonlinear internal model. In this case, the relatively large sampling time, due to the process slow rise time, gives the designer freedom in choosing the optimizer and the prediction horizon. For a faster system this task can be more difficult.

It is clear, from Fig. 7(b), that the manipulated variable given by the PID controller hit the process input constraints when set-point changes occur. On the other hand, PFC (Fig. 6(b)) seems to deal reasonably well with the input constraints. Finally, Fig. 5(b) shows that the NMPC algorithm deals with the constraints in a more optimal way, where the manipulated variable is used on its full range without violating its limits.

8. Conclusion

For the pasteurization process presented, MPC implementations give better control than the classical PID controller, even if the parameters are tuned by a computationally heavy optimization technique. There are two main reasons for that improved performance:

- Time delay is compensated, resulting in less overshoot for the MPC case. The effective elimination of the delay also allows "tighter" control without risk of instability; therefore, reducing variance.
- Constraints are handled and future real plant behaviour is predicted allowing compensation of undesirable responses before they are allowed to occur.

The linear MPC technique (PFC) is found to be best suited when a rigorous regulation around a fixed reference or set point (disturbance rejection case) is needed. The internal linearized model will then be able to predict the plant output with minor mismatch. Moreover, the PFC's suboptimal way of dealing with constraints appears to be sufficient for this case. Finally, if the plant needs rigorous control over a wide reference range with optimal constraint handling, then an online optimization MPC approach, using a nonlinear prediction model, seems best suited to sustaining the challenge at the cost of complexity and computation power.

Nomenclature

C	Controller set point
k_d	Derivative gain
k_i	Integral gain
k_p	Proportional gain
T_i	Integral time constant
T_d	Derivative time constant
T_{im}	Input milk temperature, entering section S4
T_{op}	Output milk temperature, exiting section S1
u	Manipulated variable
y_A	Auto-regressive part of the linear model
y_{ANN}	Output of the ANN model
y_F	Forced part of the linear model
y_L	Auto-regressive part of the linear model
y_M	Output of the first-order ARX model
y_P	Process output
y_R	Reference trajectory

References

[1] S.J. Quin & T.A. Badgwell, An overview of industrial model predictive control technology, in: J. Kantor, C. Garcia, & B. Carnahan (Eds.), *Chemical process control*, AIChe Symposium Series (New York: 1999), 232–256.

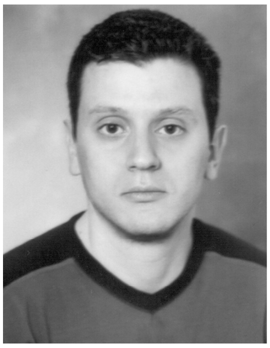
[2] J. Richalet, *Pratique de la Commande predictive*, Traite des Nouvelles Technologies, Serie Automatique (Paris: Hermes, 1993).

[3] C.R. Cutler & R.L. Rameker, Dynamic matrix control: A computer control algorithm, paper presented at the *AIChE 86th National Meeting*, April 1982.

[4] M. Morari & J.H. Lee, Model predictive control: Past, present and future, *Computers and Chemical Engineering*, *23*, 1999, 667–682.

[5] J.J. Ibarrola, J.M. Sandoval, M. García-Sanz, & M. Pinzolas, Predictive control of a high temperature short time pasteurization process, *Control Engineering Practice*, *10*(7), 2002, 713–725.
[6] G. Bylund, *Dairy processing handbook* (Tetra Pak, 1995).
[7] M.T. Khadir & J. Ringwood, Neural network modelling and predictive control of a milk pasteurization plant, *Proc. of the EANN*, Cagliari, Italy, 2001.
[8] M.T. Khadir, J. Richalet, J. Ringwood, & B. O'Connor, Modelling and predictive control of milk pasteurization in a plate heat exchanger, *Proc. of the Foodsim*, Nante, France, 2000, 216–220.
[9] Y. Le Cun, J.S. Denker, & S.A. Solla, Optimal brain damage, *Addvances in Neural Information*, *2*, 1989, 126–142.
[10] J. Sjoberg, *Non-linear system identification with neural networks*, doctoral diss., Linkoping University, Sweden, 1995.
[11] G.M. Cloarec, *Statistical methods for neural network prediction model*, Research report EE/JVR/98/1 Control System Group, School of Electronic Engineering, Dublin City University, Dublin, 1998.
[12] D.W. Clarke, C. Mohtadi, & P.S. Tuffs, Generalized predictive control, Part I: The basic algorithm, *Automatica*, *23*(2), 1987, 137–148.
[13] M.T. Khadir, *Modelling and predictive control of Glanbia Ballitoer Pasteurization Plant*, Research report EE/JVR/99/3 Control System Group, School of Electronic Engineering, Dublin City University, Dublin, September 1999.
[14] P.E Gill & W. Murray, *Minimisation subject to bounds on the variables*, Report NAC 72, National Physical Laboratory, 1976.

Biographies



Mohamed Tarek Khadir graduated from the University of Annaba, Algeria, with a State Engineering Degree in Electronics, majoring in control, in 1995. After two years' work in the computer industry, he undertook an M.Eng. at DCU, Ireland, graduating with First Class Honours in 1998. He has recently submitted a thesis for his Ph.D. degree at National University of Ireland, Maynooth (NUIM). He currently works as a Researcher and Lecturering Assistant at the Department of Electronic Engineering, NUIM. His research interests are primarily application of model predictive control to industrial processes in order to improve control performances; his latest work has been within the dairy industry. He also acts as a Consultant for the International Center for Management and Development (ICMD), Saudi Arabia, delivering courses in classical and advanced control systems in the Middle East.

John Ringwood received the B.Sc. (Eng.) and Ph.D. degrees from the University of Dublin and Strathclyde University in 1981 and 1985, respectively. He is currently Head of the Electronic Engineering Department at the National University of Ireland (NUI), Maynooth, Ireland, and Professor of Electronic Engineering. Prior to his current appointment, he was a Senior Lecturer at Dublin City University (DCU). He has also held visiting positions at Massey University, New Zealand, and the University of Auckland. His research interests lie in modelling of industrial systems and time series, control system design, and applications of control to biomedical, biotechnological, process, steel, and robotic systems. He has authored or co-authored over 80 publications in scientific journals, refereed conference proceedings, and books. He was the general chair for the 1994 Irish Signals and Systems Conference and has served as session chair or co-chair at numerous international conferences. Dr. Ringwood is a Senior Member of the IEEE, a Fellow of the IEE, a Fellow of the Institution of Engineers of Ireland, and a Chartered Engineer.