GA-Tuned Fuzzy Logic Controller for a Multiple-Effect Evaporator in the Sugar Industry

Arvin V. Pitteea, Robert T. F. Ah King and Harry C. S. Rughooputh
Department of Electrical and Electronic Engineering
Faculty of Engineering
University of Mauritius
Reduit, Mauritius
r.ahking@uom.ac.mu

Abstract – **This paper proposes a control scheme to control a Multiple-Effect Evaporator (MEE) in the sugar industry using Fuzzy Logic Control (FLC). An algorithm using Genetic Algorithms (GAs) is proposed and used to automatically tune the Scaling Factors (SFs) and Membership Functions (MFs) of the FLCs. The performances of the FLCs with only SFs tuned and with both SFs and MFs tuned are compared. The results show that tuning SFs only is sufficient to give satisfactory performance. However, the tuned MFs and termsets in general show signs of capturing the non-linearity of the plant. Significant improvement in control performance has been obtained for different control variables when MFs are tuned in addition to tuning SFs.**

I. INTRODUCTION

The multiple-effect evaporator (MEE) in use in the sugar industry is the biggest and most complex evaporation process that may be met in the industry. The economy of sugar manufacturing depends strongly on the MEE station [1] because of the huge amount of water that has to be evaporated to raise the concentration (brix) of sugar cane juice from a nominal value of 15% to syrup with a brix of 72%. The syrup is sent to the vacuum pans for sugar crystallisation. In addition, most sugar factories produce electricity from steam generated by burning bagasse, which is a by-product of sugar manufacturing. Thus, a more judicious use of steam by the MEE plant will enable more steam to be exploited in the production of electricity.

It is therefore of interest to optimise the evaporation process in a MEE in order to decrease the energy consumption but also to enhance the quality of the final product. Advanced automatic control is an important factor to achieve this [2]. The main objective to minimise the energy consumption is achieved by the development of a control scheme for the MEE in order to control the syrup brix to a high and constant value giving the maximum allowable evaporation [3]. A second control objective is to stabilise the pressure of juice steam to the vacuum pans.

In this paper, it is proposed a control scheme using simple Fuzzy Logic Controllers (FLCs) to control the MEE. A tuning algorithm is proposed to tune the FLCs based on the Genetic Algorithms (GAs). Fuzzy Logic is used essentially because the MEE is non-linear and can have varying operating points. Since FLCs are also non-linear and do not have any operating point [4], they are particularly suited to the MEE process.

This paper is organised as follows. Section II presents the MEE process. Section III presents the analytical model that is used for simulations. Section IV describes the control strategy adopted and section V the FLCs used. Section VI presents the GA tuning approach used to tune FLCs in this paper. In section VII, simulation results with FLCs with tuned Scaling Factors and Membership Functions are given and compared. Additionally, the MFs after tuning are given for comparison. Section VIII presents some conclusions and further work.

II. MULTIPLE-EFFECT EVAPORATION PROCESS

1) Principle of Multiple Effects

When juice is heated by steam in an evaporator, a quantity of juice steam, approximately equal to the amount of steam condensed in the calandria, is produced. This juice steam, which is at a lower temperature and pressure, can be utilised in turn as heating steam for a second evaporator. The juice steam from the latter evaporator can be used to heat a third evaporator and so on. This is the principle of multiple-effect. The number of effects is equal to the number of unit evaporators. To provide the necessary temperature difference for heat to flow from the first to the last effect, the last effect is connected to a vacuum pump. Thus, the pressure along the effects decreases monotonically from the first to the last.

2) Condenser

The juice steam from the last effect is usually sent to a direct contact condenser. This steam is not re-circulated in the factory and is therefore lost. For economical reasons, it is consequently recommended to minimise this steam loss as far as possible. The vacuum pump required to raise the necessary vacuum is connected to the condenser. The hot water exiting from the condenser is sent to a cooling pond where the temperature of the water is reduced before being re-circulated in the condenser.

3) Vapour Bleeding

A sugar factory is a big consumer of low pressure heating steam; the latter is required by the juice heaters to heat the raw juice coming from the mills and the clarified juice, and in the vacuum pans. There is a gradual decrease in temperature and pressure along the MEE. The MEE thus offers a complete range of vapour temperatures, which can satisfy different heating purposes. Since many heating apparatus in the factory requires only low-pressure steam, it is more economical to bleed vapour from the intermediate effects in a MEE instead of using live steam. Thus, the heaters and the MEE are dependent.

4) Presentation of MEE Station

The MEE process used in this paper simulation is depicted in Fig. 1. It consists of five effects (Roberts) with vapour bleeding from the second, third and fourth effects only. The heating steam in the first effect is exhausted from the turbo-alternator in the factory. The juice steam from the last effect is sent to a direct-contact condenser. The major part of the vapour bleeding takes place in the second effect, which supply the vacuum pans with heating steam, in addition to a juice heater.

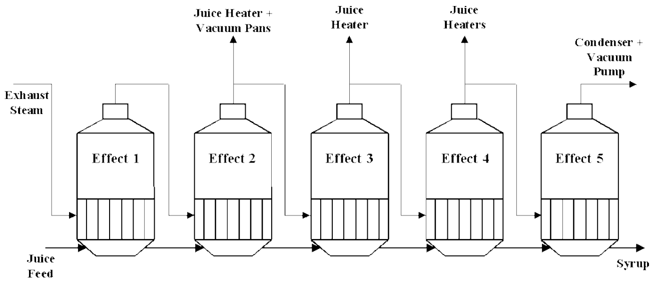


Fig. 1. Multiple-Effect Evaporator Station

Because of the large amount of juice steam consumption by the vacuum pans, and especially because of the batch nature of the vacuum pan operation, the evaporation station is frequently disturbed causing the brix to fluctuate and consequently the energy consumption of the evaporator set to increase [3]. In addition, the pressure in the second effect fluctuates a lot, which in turn affects both the pressures in the other effects and the functioning of the vacuum pans themselves.

III. Analytical Mathematical Model

Fig. 2 above shows the first two effects to illustrate the modelling. The overall model of the MEE station is obtained by the concatenation of the models of each evaporator. The model of the evaporator is built from mass and energy balances. This approach provides a lot of flexibility of developing general unit models because the model of any MEE station can be built up. However, the order of the resulting models is generally high and some form of model order reduction becomes necessary [4]. The mathematical model, as used in this paper, is given in detail in [5]. Model order reduction has been obtained by making reasonable assumptions.

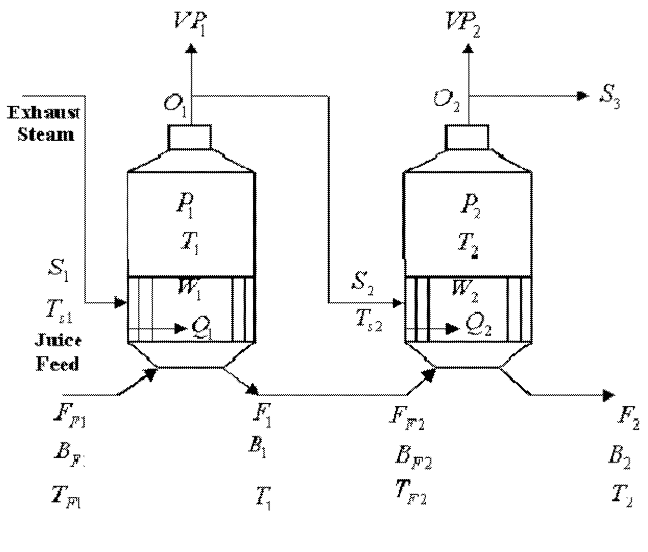


Fig. 2. First two Effects Illustrating Mathematical Modelling

The state variables are the brix and temperature of each effect and are given by (1) and (2), respectively. The delay times due to piping and /or low flow rates are neglected.

$$\frac{dB_i}{dt} = \frac{F_{Fi} \cdot (B_{Fi} - B_i) + O_i \cdot B_i}{W_i} \quad \textbf{(1)}$$

$$\frac{dT_i}{dt} = \frac{F_{Fi} \cdot \left[(h_{Fi} - h_i) - \frac{\partial h_i}{\partial B_i} \cdot (B_{Fi} - B_i) \right] - O_i \cdot \left[H_{vi} - h_i + \frac{\partial h_i}{\partial B_i} \cdot B_i \right] + Q_i}{W_i \cdot \frac{\partial h_i}{\partial T_i}} \quad \textbf{(2)}$$

where

B_{Fi}, B_i - input, output brix (mass fraction, mf)

O_i - overhead vapour flow (kg/s)

F_{Fi} - flow rate of juice/syrup into effect (kg/s)

W_i - mass of liquid hold-up inside tubes (kg)

T_i - temperature (^oC)

h_{Fi}, h_i - input, output juice/syrup enthalpy (J/kg)

H_{vi} - enthalpy of vapour (J/kg)

Q_i - heat flow from condensation to concentration chamber (J/s)

The MEE is a non-linear one and presents several control challenges, such as significant time constants and strong disturbances in the form of steam deductions. These characteristics can be observed in the step-response curves presented in Fig. 9.

Table I summarises the values of all the variables in the MEE which are obtained from [5]. Henceforth, these values will be considered as steady state values before any disturbances. The control problem, as defined earlier, shall be to maintain brix B_2 at the steady state value, B_5 at 0.72 mf and the pressure P_2 at 112 kPa in spite of any disturbances.

TABLE I

STEADY STATE VALUES

Effect, i	Temperature, T_i (°C)	Brix, B_i (mf)	Pressure, P_i (kPa)	Juice Flow, F_i (kg/s)	Steam Flow, S_i (kg/s)
1	111.6	0.1762	129.8	30.51	9.3818
2	102.9	0.2607	104.2	20.62	9.494
3	83.69	0.3587	53.11	14.99	5.021
4	72.57	0.5041	32.55	10.67	4.073
5	55.05	0.7178	13.96	7.490	2.899

IV. Control Strategy

A. Decomposition of MEE

After obtaining a plant model adequate for simulation, it is now necessary to develop control strategies that will enable the control of the MEE. The goal of the overall control system is to minimise the energy consumption of the MEE and enhance product quality. This is achieved by minimising fluctuations in the brix of the syrup (B_5) and the steam pressures, especially in the second effect.

The deduction of steam at the second effect affects strongly the plant. This disturbance is reflected quite fast in the pressure P_2 and the brix B_2. To enable the vacuum pans to work satisfactorily at all times, the steam pressure P_2 must be maintained sufficiently high. In addition, regulating the brix B_2 minimises the propagation of disturbances down the plant [3] and hence enables a better control of the brix of the syrup (B_5) at the fifth effect.

The MEE is a multivariable plant. It is proposed in this paper to use Single Input Single Output (SISO) fuzzy controllers that shall synergistically control the whole MEE. It is therefore necessary to decompose the process into subsystems and to identify the inputs (manipulated variables + disturbances) and the outputs (controlled variables). From the viewpoint of control engineering, the process has three inputs, namely the steam flow and the juice flow into the first effect, and the juice steam flow out of the fifth effect. The outputs are the pressure P_2, brix B_2 and the brix B_5. The main disturbance considered is the variation of the steam flow to the vacuum pans.

To identify the SISO control loops, a 10% step increase and decrease was applied to the four inputs and the effects on the three outputs was observed. It was found that both the inputs S_1 and VP_2 affect strongly the pressure P_2. On the other hand, it was found that the brixes B_2 and B_5 are mainly affected by the juice feed F_{F1} and the juice steam flow O_5, respectively. It was therefore chosen to control the pressure P_2 by manipulating the steam flow S_1 and to control the brix B_2 by varying the juice feed F_{F1}. The brix B_5 is controlled by manipulating the speed of the vacuum pump, which results in the variation of the juice steam flow O_5 to the condenser. Thus, the process is divided into two main subsystems, namely effects 1–2 and effects 3–5.

B. Control Architecture

Fig. 3 illustrates the control architecture for effects 1-2. There are three fuzzy PI controllers. The fuzzy PI *Pressure2* controller is in cascade with the *Steam* controller. This arrangement is chosen because the steam supply pressure can vary in practice and thus the steam controller will maintain the setpoint provided by the *Pressure2* controller. The steam controller is in fact a servo controller whereas the Pressure2 controller is a regulatory controller because they track and maintain setpoints, respectively. The juice feed is manipulated as a function of the steam flow by a ratio station, the ratio being provided by the *Brix2* controller. PI type controllers are used for the three controllers because it is desired to maintain the pressure P_2 and the brix B_2 at setpoint with zero steady-state error.

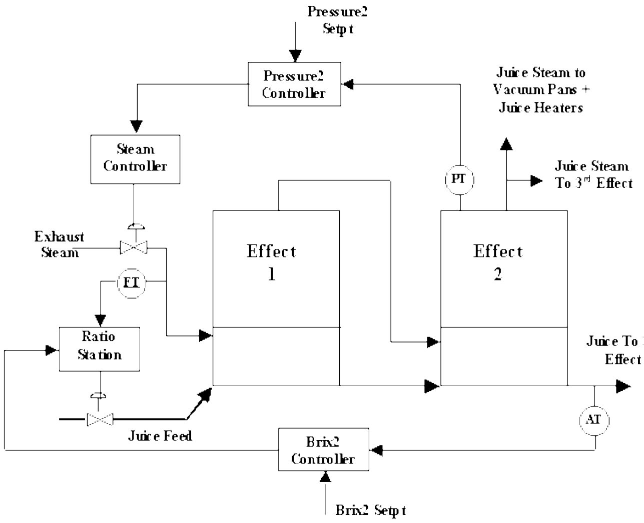


Fig. 3. Control Scheme for effects 1-2.

Fig. 4 depicts the control concept at the last effect. It consists of a *Brix5* feedback controller, which is of the fuzzy PI type. To enhance the accuracy of the brix B_5 control, feedforward control is included to compensate for the two major disturbances entering the fifth effect, namely the variations of effect4 juice flow and effect4 brix. Since these feedforward controllers do not maintain any setpoint, they have been chosen to be of the PD type; the D part of the controller enables a better tracking of the disturbances and hence acts in a more anticipatory way to compensate for the disturbances. These feedforward controllers are essential considering the significant time constant of the fifth effect.

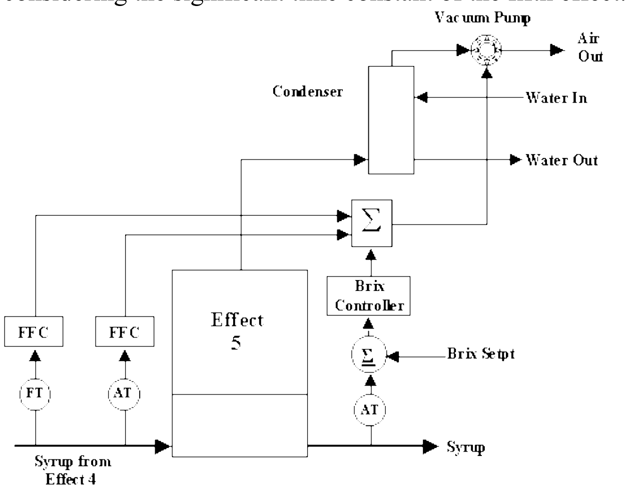


Fig. 4. Control Scheme for Effect 5

V. Fuzzy Logic Controllers (FLC's)

Several FLCs were developed to synergistically control the MEE plant (Fig. 3 and 4). Since the plant is MIMO, and since fuzzy logic enables the design of MIMO controllers, only one FLC could have been designed, in principle, to control the whole MEE. However, then the development of the control heuristics and the management of the rules would have been problematic. Furthermore, with distributed FLCs, it is easier to access the contribution of a particular controller.

Two main approaches exist for the design of FLCs. In the first one, which is adopted by [6] and in [7], the limits of the *universe of discourse* (UOD) of the termsets is set according to the maximum range between which the input signals may vary. In the second approach, which we shall adopt in this paper, the UOD is normalised between –1 and 1 and scaling factors (SFs) are placed on the inputs and outputs of the FLC. Thus, the signals entering and leaving the FLC are per unit variables. The advantage of fuzzy control in terms of per unit variables is that the same control algorithm can be applied to all plants of the same family. Besides, it becomes convenient to design the FLC [8].

Thus, a FLC is composed of SFs, termsets containing *Membership Functions* (MFs) and a rule base. The latter contain the knowledge about how to control the plant, which can be obtained from an expert, plant operators or common sense. All the rule bases in this paper are developed from common sense. This is the main advantage of using simple, distributed FLCs.

Tuning of a FLC involves modifications of the SFs, MFs and the rule base, either alone or in combination. However, the order in which these three components are tuned is very important [9]. In this work, we begin with macroscopic effects, by tuning the SFs, while using a standard, normalised, uniformly spread termsets and a homogeneous rule base. After obtaining near optimal SFs, we proceed to tune the termsets causing medium-size effects. Additionally, the rule base may be tuned to achieve microscopic effects. However, in this paper, attempt shall not be made to tune the rule base.

The *Steam*, *Pressure2*, *Brix2* and *Brix5* controllers are of the PI type whereas the *FlowFeedforward* and *BrixFeedforward* controllers are of the PD type. The structures of the fuzzy PI and PD controllers used in this paper are shown in Fig. 5. The type of PD controller is chosen as shown because its output signal before integration is incremental in nature just like the PI controller and thus the outputs from both the feedback and the feedforward controllers in Fig. 4 can be summed together before being integrated to generate the total control signal that is sent to the vacuum pump. Furthermore, this arrangement of the feedback and the feedforward controllers naturally enables a kind of "bumpless transfer" [10] when feedforward control is enabled.

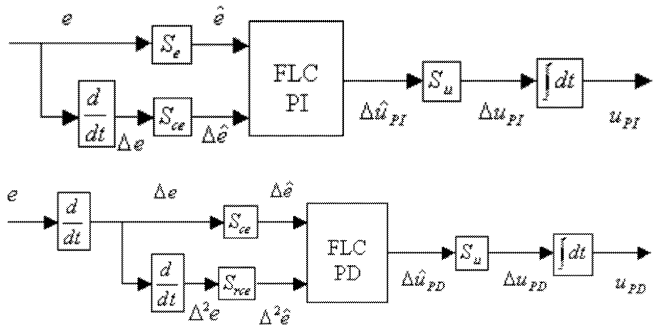


Fig. 5. Structures of PI and PD Controllers used

All the FLCs have two inputs and one output. The input termsets have five triangular MFs while the output termsets have seven triangular MFs. All the FLCs have normalised uniformly spread termsets as shown in Fig. 6.

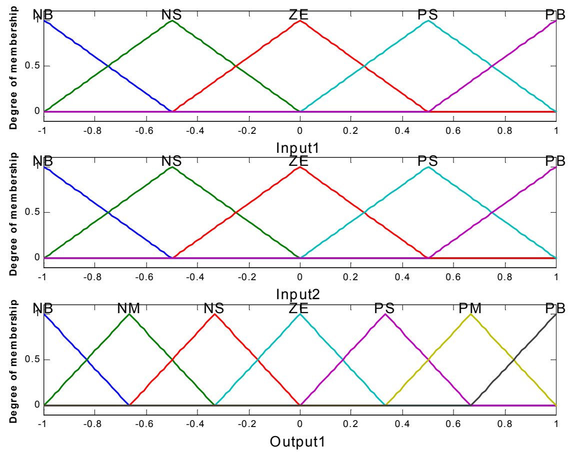


Fig. 6. Membership Functions of FLCs before Tuning

The rule base, however, differs among the FLCs. The rule base for the *Steam*, *Pressure2*, *FlowFeedforward* and *Brix5* controllers are presented in Table IV(a) .Table IV(b) presents the rule base for the *Brix2* and the *BrixFeedforward* controllers. Input1 and Input2 in Table IV(a) are, respectively, the error and derivative of the error for the *Steam* and *Pressure2* controllers, whereas they are the derivative and the double derivative of the juice flow for the *FlowFeedforward* controller. Input1 and Input2 in Table IV(b) are, respectively, the error and derivative of the error for the *Brix2* controller, whereas they are the derivative and the double derivative of the brix4 for the *BrixFeedforward* controller. The defuzzyfication method used was the centre of area for all controllers.

TABLE IV: RULE BASE FOR FLCS. (A) STEAM, PRESSURE2, FLOWFEEDFORWARD AND BRIX5, (B) BRIX2 AND BRIXFEEDFORWARD

Output		Input1				
		NB	***NS***	***ZE***	***PS***	***PB***
Input2	***NB***	NB	NB	NM	NS	ZE
	NS	NB	NM	NS	ZE	PS
	ZE	NM	NS	ZE	PS	PM
	PS	NS	ZE	PS	PM	PB
	PB	ZE	PS	PM	PB	PB

Output		Input1				
		NB	***NS***	***ZE***	***PS***	***PB***
Input2	***NB***	PB	PB	PM	PS	ZE
	NS	PB	PM	PS	ZE	NS
	ZE	PM	PS	ZE	NS	NM
	PS	PS	ZE	NS	NM	NB
	PB	ZE	NS	NM	NB	NB

VI. AUTOMATIC TUNING OF FUZZY CONTROLLERS USING GA

A. Genetic Algorithms (GA)

Genetic Algorithms (GAs) are a family of computational models inspired by evolution [11]. They are stochastic global search and optimisation methods that mimic the metaphor of natural biological evolution. GA's operate on a population of potential solutions applying the principle of survival of the fittest to produce successively better approximations to a solution. At each generation of a GA, a new set of approximations is created by the process of selecting individuals according to their level of fitness in the problem domain and reproducing them using operators borrowed from natural genetics. This process leads to the evolution of populations of individuals that are better suited to their environment than the individuals from whom they were created are, just as in natural adaptation.

The automatic tuning of the FLCs is done in two stages. In the first stage, the GA finds the optimum values of the SFs. In the second stage, the GA adjusts the position and shape of the MFs that compose the linguistic variables used by the controllers. To use GAs to automatically tune the FLCs, it is necessary to define a codification for the controller parameters, and a fitness function.

B. Codification of the FLC for Tuning by GA

1) Scaling Factors: The potential SFs of a FLC are concatenated as a string of real numbers to form a chromosome. Since the requirements in terms of programming are relatively inexpensive when using the GA to tune SFs, the SFs of several FLCs may be tuned simultaneously. In this work, the SFs of two batch of FLCs were tuned simultaneously, namely those in Fig. 3 and in Fig. 4.
2) Membership Functions: Fig. 7 shows a normalised termset containing seven uniformly positioned triangular MFs that will illustrate the tuning strategy. Each MF is defined by three parameters, P_1, P_2 and P_3, where

$$P_1 \le P_2 \le P_3 \qquad \textbf{(3)}$$

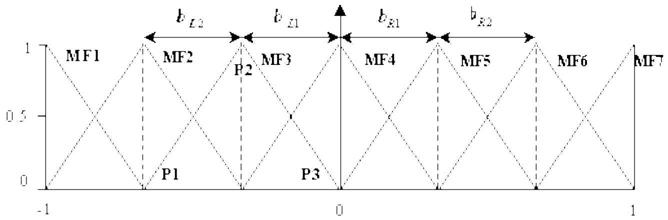


Fig. 7. Uniformly spread termset to illustrate tuning strategy

is a necessary requirement in order not to violate the definition of a triangular MF.

Tuning of MFs usually consists of moving the MFs across the UOD such that a chosen performance criterion is improved. If the SFs are priory optimally tuned, then at a particular operating point, the input signals are properly mapped onto the UOD. As such, it is assumed, in the proposed algorithm, that it is not necessary to change the parameters that are coincident with the extreme ends of the UOD in any termsets. In addition, the structure of most control problem suggests that the centre point of the middle MF lies at the origin itself. Thus the second parameter of the first, middle and last MF will never be changed as will the first parameter of the second MF and the third parameter of the before last MF. All other parameters of any MF will then be changed during the tuning process.

In order to use GA to tune the MFs, some parameter that reflects the movement of the MFs must be identified. Different scenarios can be envisaged for tuning the MFs. However, in the proposed algorithm, it is assumed that the peaks and valleys, which constitute the termsets, are moved together to and fro along the UOD. Thus, it can clearly be seen that the length of the intervals between the peaks is an appropriate parameter that reflects the movement of the MFs when the latter are tuned.

For the measurement of the intervals, some reference is required. In the present algorithm, the intervals are measured with respect to the middle MF (MMF). Thus, it is mandatory to have an odd number of MFs in any termsets. In fig. 7, for 7 MFs, there are 3 intervals on either sides of the peak of the MMF. However, only 2 intervals are considered to be included as genes in the chromosome for the GA since the third interval is not independent of the other two; a consequence of an earlier assumption. Thus the number of tuneable intervals is equal to the number of peaks and is given as

$$\#(Int) = \#(MF) - 3 \qquad \textbf{(4)}$$

where

$\#(Int)$ is the number tuneable intervals

and $\#(MF)$ is the number of MF.

Since the position of the MMF is critical for the algorithm, the identification of that position for any FLC with any input or output termset with any number of odd MFs is crucial. The index of the MMF is thus given as

$$j = \frac{\#(MF)+1}{2} \qquad \textbf{(7)}$$

where

j = index of middle MF.

The length of the intervals are given in general as

$$b_{Li} = \left| P_{(j-(i-1))2} - P_{(j-i)2} \right| \qquad \textbf{(5)}$$

$$b_{Ri} = \left| P_{(j-(i-1))2} - P_{(j+1)2} \right| \qquad \textbf{(6)}$$

where P = parameter of MFs and i is the interval number given as

$$i = \frac{\#(Int)}{2}. \qquad \textbf{(7)}$$

The first subscript (in parentheses) in Eq. (5) and (6) is the index of a MF while the second subscript is the second parameter of a triangular MF (which is always 2).

Equations (4) to (7) and inequality (3) are used at the beginning of the MF tuning process after the SFs are optimally tuned to read the initial termsets and transform the parameters of the MFs into the genes of the chromosome. This step seeds the initial population with an individual that already gives satisfactory performance and thus enable the GA to concentrate its search in fine-tuning the FLC. This prevents precious processing time from being wasted. Furthermore, the proposed tuning algorithm preserves the order of the MFs from left to right, thus rendering the optimisation using the GA more efficient by *not* trying possibilities that logically would be impossible.

For the GA to randomly generate new genes defined by (5) and (6), the intervals within which these parameters may lie must be given. As seen in fig. 7, the range of values that the intervals can take, must not allow the sum of the intervals to exceed 1. Thus

$$\sum_i b_{Li} \le 1 \qquad \textbf{(8)}$$

and

$$\sum_i b_{Ri} \le 1 \qquad \textbf{(9)}$$

In general,

$$\min\left(b_{Li}\right) = \min\left(b_{Ri}\right) = 0 \quad \textbf{(10)}$$

$$\max\left(b_{Li}\right) = \max\left(b_{Ri}\right) = \frac{\max Range - \min Range}{\#\left(Int\right)} \quad \textbf{(11)}$$

where *Range* is the limits within which the UOD is defined. Eqs (10) and (11) calculates the range within which the GA can randomly generate the values of the genes for each individual at every generation.

After each generation of evolution, the GA comes up with a population of potential FLCs whose performance must be assessed. Unlike the tuning of SFs, where the genes of the chromosomes were simply the SFs themselves and thus meaningful FLC parameters, in the proposed algorithm, the genes in the chromosomes do not have a direct meaning as far as the parameters of the FLCs are concerned. It is therefore required to perform a reverse mapping to convert the chromosome into the parameters (P_1, P_2, P_3). This is achieved by recognising that the length of an interval simultaneously influence the parameter of three MFs, as can clearly be seen in fig. 7. Thus, we have

$$P_{(j-(i-1))1} = P_{(j-i)2} = P_{(j-(i+1))3} = b_{Li} \quad \textbf{(12)}$$

$$P_{(j+(i+1))1} = P_{(j+i)2} = P_{(j+(i-1))3} = b_{Ri} \quad \textbf{(13)}$$

Equations (12) and (13) are solved in a loop for each interval and for all input and output termsets that a FLC may have. It should be noted that the implementation of the above algorithm involves more algorithmic complexity that depends also on the data structures used. Nonetheless, the above equations illustrate the essence of the tuning idea and algorithm.

Contrary the other works on the automated tuning of FLC [6], [9], it is not the parameters defining the MFs that are coded but the distances between the peaks. This idea has also been used in [12]. In the paper, the evolution strategy presented only tune the centre points (P_2) of the triangular output MFs. This means that all the MFs can be moved. The proposed algorithm in this paper, in contrast, tunes all three parameters, but not of all MFs. In [12], the shape of the MFs was restricted to symmetric triangles of fixed width whereas this restriction is not imposed in the proposed algorithm. In addition, in the proposed tuning strategy, the left and right MFs can be tuned independently and thus this creates asymmetric termsets, which, in addition to asymmetric triangular MFs, may be useful in certain control problem where plants are very non-linear. A further difference is that in the presented algorithm, it is the peak of the MMF that is taken as reference to measure the distances, whereas in [12], it is the minimum range of the UOD that is taken as reference.

A major limitation of the proposed algorithm is that the peaks and valleys are moved together. In a more advanced algorithm, they may be moved independently. However, this is very challenging in order to satisfy eq. (3).

All the genes in the chromosomes are coded as real numbers for tuning both the SFs and the MFs [13], [14].

C. *The Objective Function*

To evaluate the effectiveness of each individual in a population, the process simulation has been used. The controller acted over the simulated system, trying to maintain the controlled variables P_2, B_2 and B_5 at setpoint in the event of disturbances. The smaller the error between the reference and the actual value of the control variable, the better fit would be considered an individual. To put more pressure on the GA to come up with controllers with good steady state performance, the time factor is included [10]. To enable comparison between the effectiveness of the control of the different control variables since the magnitude of the errors of the pressure and brix are different, the error is normalised. To render the objective value independent of the simulation duration, a mean is taken by dividing the normalised ITAE value by the simulation time. Thus, the objective function is defined as the minimisation of the mean normalised ITAE criterion

$$f_{obj} = \min\left(\frac{\int_{t_0}^{t_f}\left|\frac{{}^{*}CV - CV}{{}^{*}CV}\right|\cdot t\, dt}{t_f - t_0}\right) \quad \textbf{(2)}$$

where t_0 and t_f are the initial and final time, respectively, ${}^{*}CV$ and CV are the setpoint and actual values, respectively, of a controlled variable, and t is the time step.

D. *Disturbance Pattern*

The major disturbance consisting of steam deductions at the second effect has been considered in this paper to be the only disturbance affecting the plant. Non-linear systems respond differently in terms of steady state outputs to step increase and decrease of equal magnitudes in inputs, and this applies to evaporators too [4]. The algorithm to tune MFs presented earlier has the capability to create both asymmetric triangular MFs and asymmetric termsets and this enables the fuzzy controller to better capture the non-linearity of the plant. However, for the non-linearity of the MEE to be manifested to the fuzzy controllers, the plant must be forced to respond in such a way that the controlled variables both increase and decrease significantly about the steady state values. To achieve this, the disturbance pattern for the steam deductions at the second effect has been chosen as $\pm$30% of the steady state value as shown in Fig. 8.

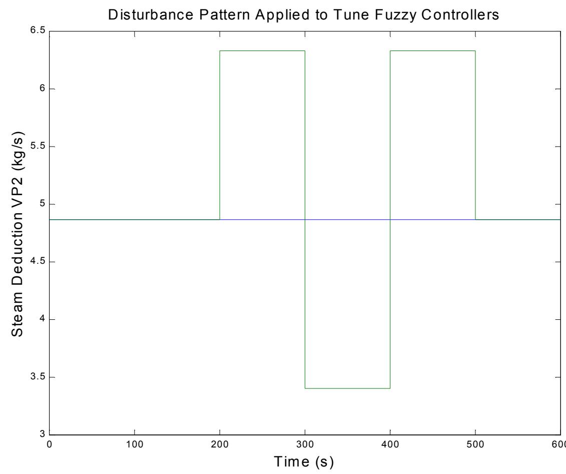
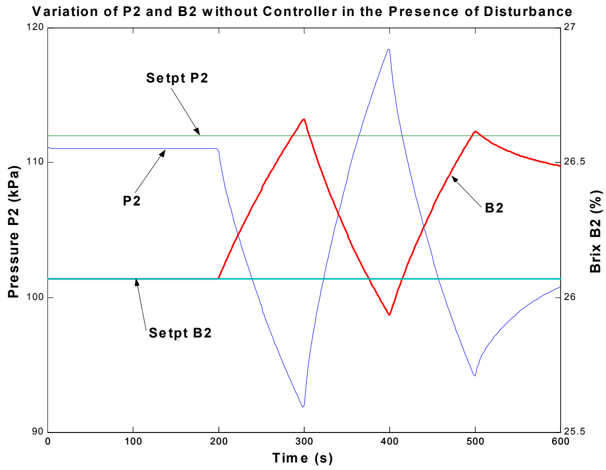


Fig. 8. Disturbance Pattern for Steam Deduction at Second Effect used for Tuning Fuzzy Controllers

VIII. RESULTS

A. *Performance without Controller*

The MEE plant was simulated with disturbance pattern applied without any controller and with all variables at steady state. The behaviour of the plant concerning the controlled variables P_2, B_2 and B_5 is shown in Fig. 9. It is clear that in the presence of disturbances, both the pressure and the product quality are not maintained at setpoint.



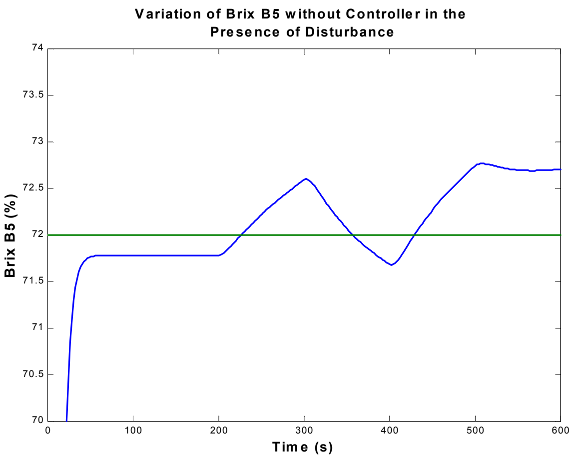


Fig. 9. Variation of P2, B2 and B5 without controller with Variation in Steam Deductions at Second Effect.

B. *Tuning of Steam, Pressure2 and Brix2 Controllers*

1) GA Tuning of SFs

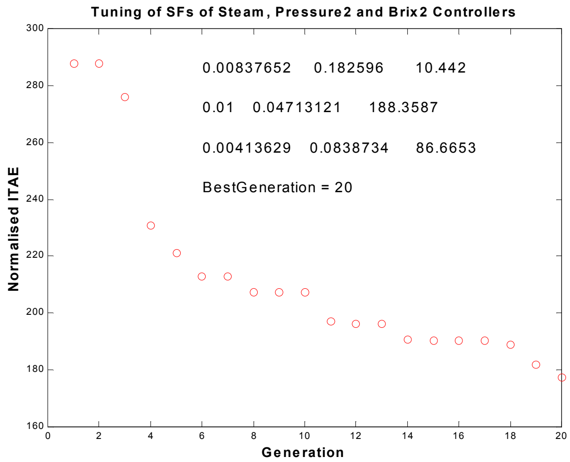


Fig. 10. Evolution of Best Individual when tuning SFs of Steam, Pressure2 and Brix2 Controllers

Fig. 10 shows the evolution of the tuning of the SFs of the three controllers over 20 generations.

2) GA Tuning of MFs

The MFs of the *Pressure2* controller were tuned first followed by those of the *Brix2* controller. The evolution of the tuning for the Pressure2 and Brix2 are shown in Fig. 11 and 12, respectively.

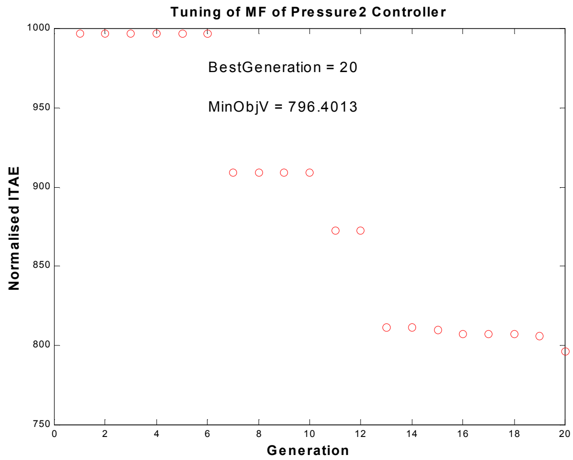


Fig. 11. Evolution of Fitness value of Best Individual when Tuning MF of Pressure2 Controller

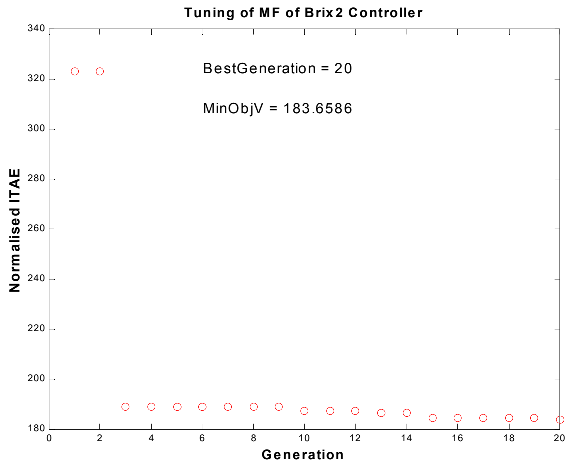


Fig. 12. Evolution of Fitness value of Best Individual when Tuning MF of Brix2 Controller

The control performance of the pressure P_2 with only SFs tuned and with both SFs and MFs tuned is shown in Fig. 13. As can be seen, there is less overshoot and faster return to reference at some point. There is a reduction of 16.8% in fitness value with tuned MF, which is significant. However, it can be observed that more improvement is still possible.

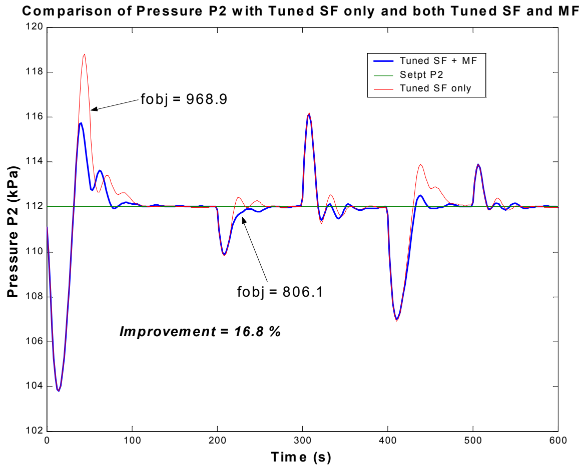


Fig. 13. Variation of P2 with controllers having both SFs and MFs tuned

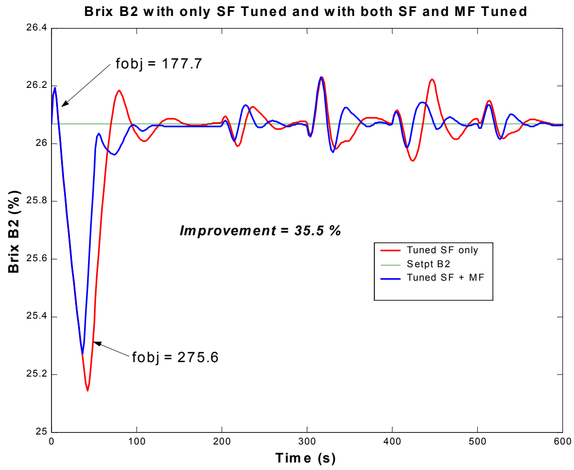


Fig. 14. Brix2 control with and without tuned MF

Fig. 14 shows the control of brix B_2 with controller having only tuned SFs and both SFs and MFs tuned. The reduction in fitness value is 35.5%, which is even more dramatic. Fig. 14 shows that the *Brix2* controller with tuned MFs acted faster in a more anticipatory way. In addition, the return to setpoint was faster with less oscillation.

Fig. 15 and 16 show the tuned MFs of the Pressure2 and Brix2 controllers, respectively.

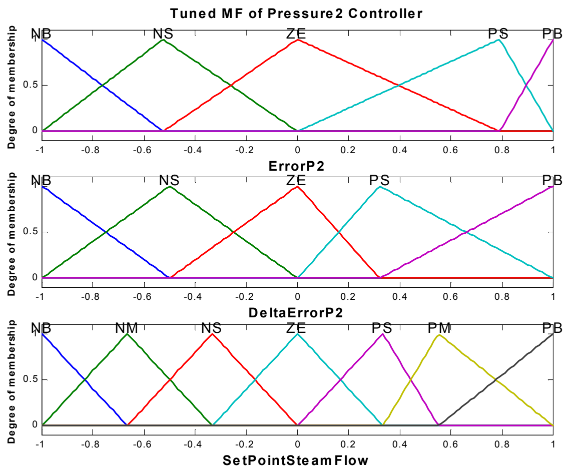


Fig. 15. Tuned MF of Pressure2 Controller

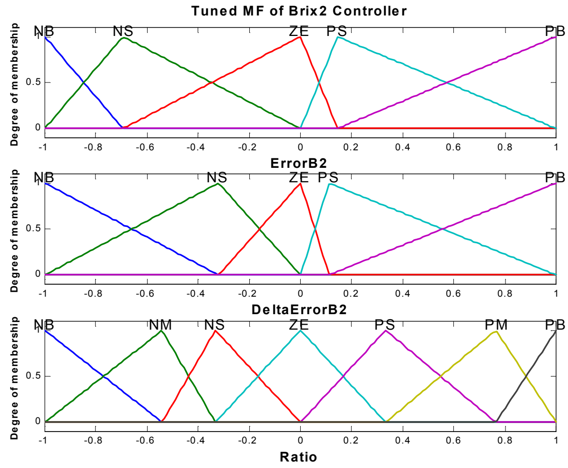


Fig. 16. Tuned MF of Brix2 Controller

C. Tuning of Brix5, BrixFeedforward and FlowFeedforward Controllers

1) GA Tuning of SFs

Fig. 17 shows the evolution of the tuning of the SFs of the three controllers over 20 generations.

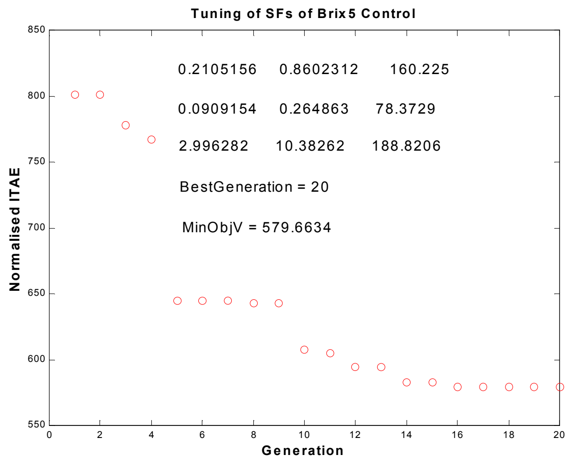


Fig. 17. Evolution of Fitness Value of Best Individual when tuning SFs of Brix5, BrixFeedforward and FlowFeedforward Controllers

2) GA Tuning of MFs

The MFs of the Brix5 controller were tuned first, followed by those of the BrixFeedforward and FlowFeedforward controllers. After the MF of one controller has been tuned, the controller is replaced with the tuned one before the MFs of the next controller is tuned. The evolutions of the tuning of the three controllers are shown in Fig. 18 - 20.

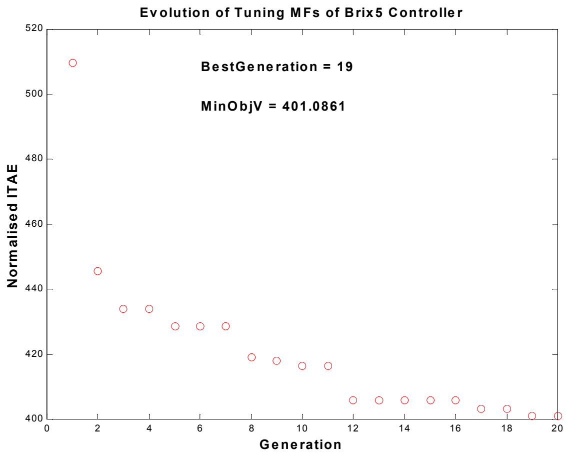


Fig. 18. Evolution of Fitness value of Best Individual when Tuning MF of Brix5 Controller

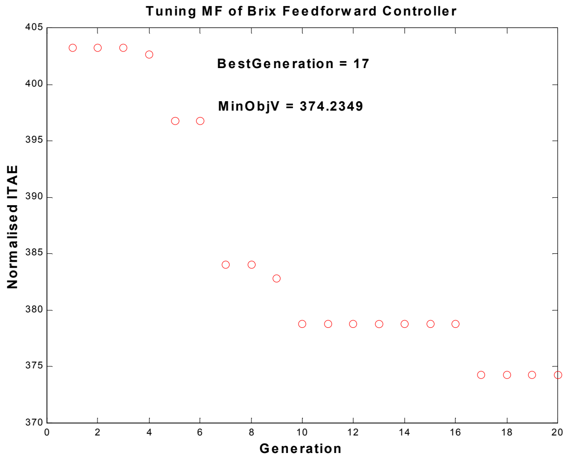


Fig. 19. Evolution of Fitness value of Best Individual when Tuning MF of BrixFeedforward Controller

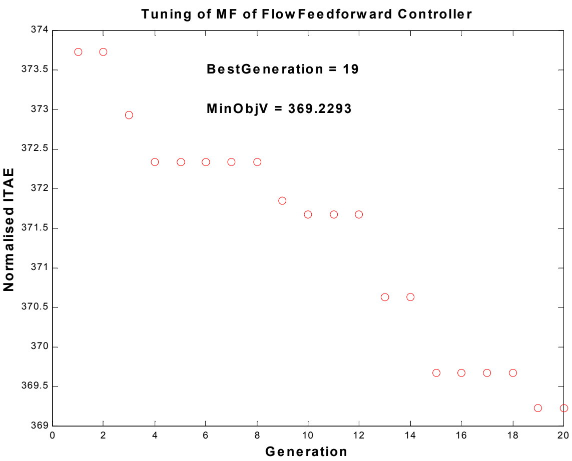


Fig. 20. Evolution of Fitness value of Best Individual when Tuning MF of FlowFeedforward Controller

Figs. 21, 22 and 23 show the tuned MFs of the *Brix5*, *BrixFeedforward* and *FlowFeedforward* controllers, respectively. It can be observed that there is almost no change in the MFs of the *FlowFeedforward* controller. This is because the tuning of the MFs of the *Brix5* controller has already much improved the control accuracy of the brix B_5 as evidenced in Fig. 24 by smaller overshoot and undershoot. In addition, with tuned MF of the *Brix5* controller, the brix5 variations remain closer to setpoint. The reduction in fitness value with Brix5 tuned MFs is 30.8% whereas the tuning of the MFs of the *BrixFeedforward* controller only reduced the fitness value a further 6.7%. The further reduction in fitness value when tuning the *FlowFeedforward* controller was only a small 1.3%. The actual Brix5 variations using manual control measured at the sugar factory over a one week running period is shown in Fig. 25. It is clear from comparison of Figs. 24 and 25 that a marked improvement in performance is achieved using the proposed method.

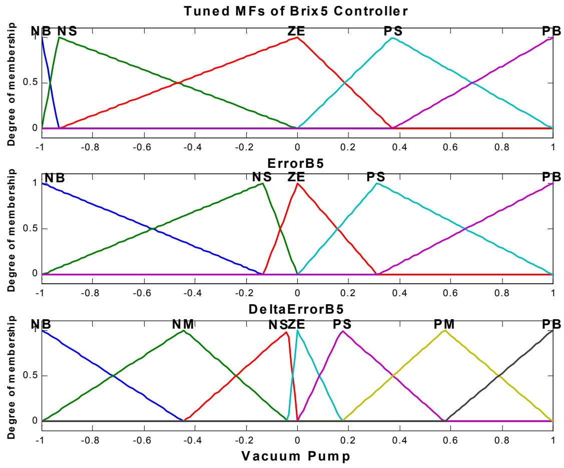


Fig. 21. Tuned MF of Brix5 Controller

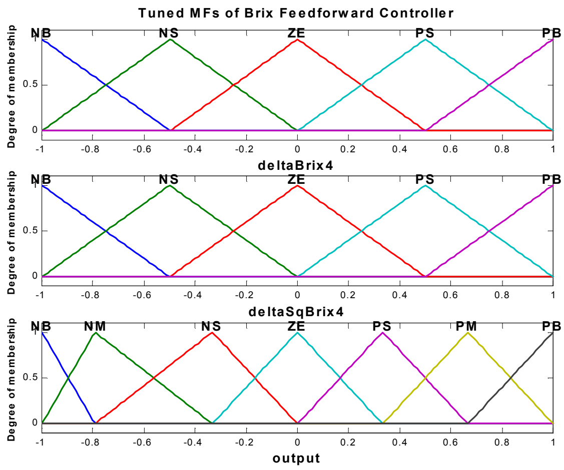


Fig. 22. Tuned MF of BrixFeedforward Controller

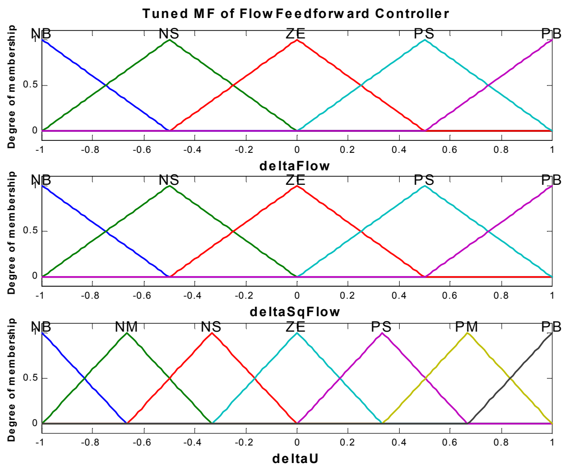


Fig. 23. Tuned MF of FlowFeedforward Controller

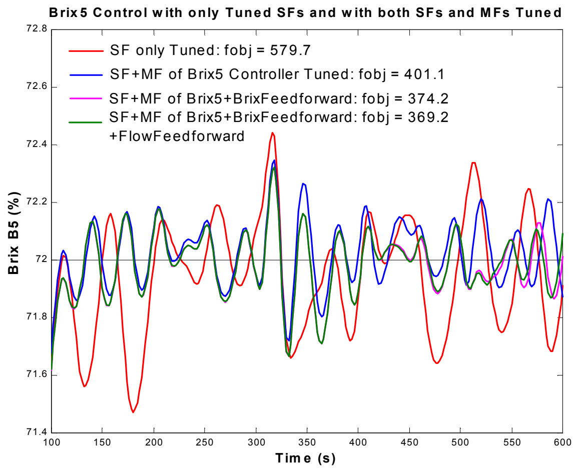


Fig. 24. Brix5 Variations

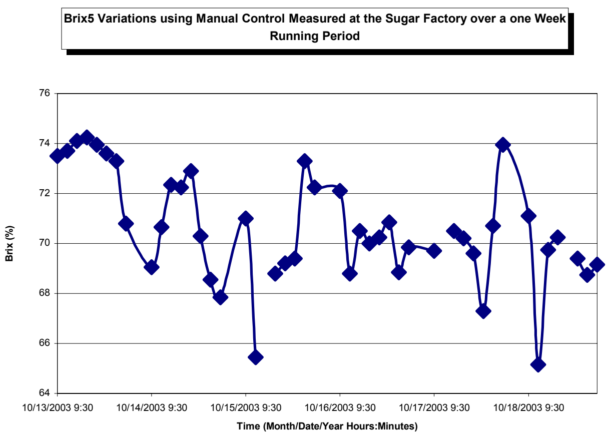


Fig. 25. Actual Brix5 Variations using Manual Control

VIII. Conclusions

An analytical model of the MEE was used to implement fuzzy controllers that were to synergistically control the whole MEE. GAs were used to tune both the SFs and the MFs. An algorithm was proposed to tune triangular MFs that retain the integrity of the MFs while at the same time enabling the MFs to be tuned asymmetrically and independently, thus enabling the fuzzy controllers to better deal with non-linear systems. The results showed substantial improvement in control accuracy even after the SFs are tuned. In general, it was found that the controlled variables pressure2, brix2 and brix5 were maintained at setpoint accurately enough in spite of disturbances.

Nevertheless, further experimentation must be done to assess the impact of the GA parameters, such as the number of generations, on the fitness value since it is observed that further improvement is possible. In addition, in the future, work must be carried out concerning the effect of tuning the MFs of several FLCs simultaneously on the results.

Concerning the control scheme, the juice processing rate by the MEE will be determined the juice steam consumption by the juice heaters and vacuum pans. The only way to force the MEE to process more juice is to change the brix2 setpoint. In the future, the automatic adjustment of this setpoint with the idea of optimising the MEE will be considered.

References

[1] S. Lissane Elhaq, F. Giri, H. Unbehauen, " The Development of Controllers for a Multiple-Effect Evaporator in Sugar Industry", http://www.cds.caltech.edu/conferences/related/ECC97/proceeds/751_1000/ECC837.PDF

[2] Alberto Lebre Cardoso and Antonio Dourado, "A Reduced Order Model and a Robust Controller's Synthesis of a Black Liquor Evaporation System", Proc. IFAC Conference on System Structure and Control, Nantes, France, July, 8-10, 1998, pp. 807-812.

[3] K. M. Nielsen, J. F. D. Nielsen, T. S. Pedersen, " Simulation and Control of a Multiple Effect Evaporator", http://www.control.auc.dk/preprint/?action=abstract&abstract=4147.

[4] Robert B. Newell and D. Grant Fisher, " Model Development, Reduction, and Experimental Evaluation for an Evaporator", Ind. Eng. Chem. Process Des. Develop., Vol. 11, No. 2, 1972.

[5] Arvin V. Pitteea, Robert T. F. Ah King and Harry. C. Rughooputh, "Parameter Estimation of a Multiple-Effect Evaporator by Genetic Algorithms", submitted to SCIS2004.

[6] Joao A. Fabro and Lucia V. R. Arruda, " Fuzzy-Neuro Predictive Control, Tuned by Genetic Algorithms, Applied to a Fermentation Process", Proceedings of the 2003 IEEE International Symposium on Intelligent Control, Houston, Texas, October 5-8, 2003, pp. 194-199.

[7] V. A. Constantin, "Fuzzy Logic and Neurofuzzy Applications Explained", Prentice Hall PTR, 1995.

[8] Bimal K. Bose, "Modern Power Electronics and AC Drives", Prentice Hall PTR, 2002.

[9] Piero P. Bonissone, Pratap S. Khedkar, Yuto Chen, " Genetic Algorithms for Automated Tuning of Fuzzy Controllers: A Transportation Application", http://www.rpi.edu/~bonisp/fuzzy-course/Papers-ps/train.ps

[10] Carlos. A. Smith, Armando B. Corripio, " Principles and Practice of Automatic Process Control", 2nd edition, John Wiley & Sons, Inc, 1997.

[11] Darrel Whitley, " A Genetic Algorithm Tutorial", http://samizdat.mines.edu/ga_tutorial/ga_tutorial.ps

[12] Frank Hoffmann, "Evolutionary Algorithms for Fuzzy Control System Design", *Proceedings of the IEEE*, Vol. 89, No. 9, September 2001, pp. 1318-1333.

[13] The MATLAB Genetic Algorithm Toolbox v1.2 User's Guide http://www.shef.ac.uk/~gaipp/ga-toolbox/manual.pdf

[14] A.J. Chipperfield, P. J. Fleming, " The Matlab Genetic Algorithm Toolbox", http://www.shef.ac.uk/~gaipp/ga-toolbox/intro_paper_1.pdf