©2010 International Journal of Computer Applications (0975 – 8887)
Volume 1 – No. 10

Evolutionary Design of Intelligent Controller for a Cement Mill Process

Dr.P. Subbaraj
Director Research
Arulmigu Kalasalingam College of Engineering ,
Tamilnadu, India

P.S. Godwin Anand
Research Scholar
Anna University,Chennai

ABSTRACT
The Knowledge Base of a Fuzzy Logic Controller (FLC) encapsulates expert knowledge and consists of the Data Base (membership functions) and Rule-Base of the controller. Optimization of these Knowledge Base components is critical to the performance of the controller and has traditionally been achieved through a process of trial and error. Such an approach is convenient for FLCs having low numbers of input variables however for greater number of inputs, more formal methods of Knowledge Base optimization are required. Genetic Algorithms (GAs) provide such a method to optimize the FLC parameters. An intelligent multi input multi output (MIMO) control for the cement milling circuit is presented. The FLC is optimized by GA for varying nonlinearity in the plant. The proposed control algorithm was tested on the cement mill simulation model within MATLABTM SimulinkTM environment. Parameters of the simulation model were set up based on the actual cement mill characteristics. The performances of the proposed control technique are compared with various control technique. The results of the control study indicate that the proposed algorithm can prevent the mill from plugging and control the cement mill circuit effectively compared to the other control technique.

Keywords
FLC, GAs, Optimization, Cement Mill, plugging

1. INTRODUCTION
Many industrial process systems may not be as readily described mathematically due to the complexity of the components of the plant and the interaction between them. Cement mills are complex processing systems with interconnected processing and drive operations [1]. It is well known that material grinding depends on many factors including mill geometry, speed, ball size distribution, mineral grindability and granule geometry. Due to the inherent process complexivity development of an accurate model of the cement milling circuit is not a simple task [2]. On some occasions, it is observed on real plants that intermittent disturbances like instance changes in the hardness of the raw material may drive the mill to a region where the controller cannot stabilize the plant. This is well understood by the operators as the so-called plugging phenomenon of ball mills [3][4].

Multivariable control techniques based on Linear Quadratic Control theory have been introduced to improve the performances of the milling circuit [5]. However, this controller, whose design is based on a linear approximation of the process, is only effective in a limited range around the nominal operating conditions. The design parameters of the LQG controller are still chosen by a trial and error method [6]. A recent contribution to the cement milling circuit control focuses on a multivariable nonlinear predictive control technique. Although this technique gives satisfactory performance in terms of robustness and stability, the design of the controller depends strictly on the mathematical model of the plant.[7],[8].

The expert system is the most appropriate solution, in most cases the fuzzy version give better results than the classical one [6]. The variety of fuzzy control applications indicates that this technique is becoming an important tool for complex processes [9]. Fuzzy control is a promising new way to face complex process control problems and the tendency is to increase their range of applicability in industrial processes [10]. Although expert-system-based solutions are effective in controlling the processes, this methodology has inherent limitations, since it is designed to mimic a human operator with inherent decision-making limitations[11]. In the absence of such knowledge, a common approach is to optimize these FLC parameters through a process of trial and error [12]. This approach becomes impractical for systems having significant numbers of input since the rule-base size grows exponentially and consequently the number of rule combinations becomes significantly large [13]. The use of Genetic Algorithms (GA) in this regard can provide such solutions [14], [15], [16]. Genetic Algorithms (GAs) [17] are robust, numerical search methods that mimic the process of natural selection. Although not guaranteed to absolutely find the true global optima in a defined search space, Genetic fuzzy systems are capable of dealing with the curse of dimensionality for complex problems with high dimensionality [18].

In this paper, a control scheme which optimizes the rule base and membership function of a MIMO fuzzy logic control for a nonlinear model of the cement mill circuit is presented. The optimization is done by GA based on minimization on Integral absolute error (IAE) of finished product y_f and mill level z. The performance of the proposed control scheme is tested for different setpoints of y_f, z and hardness parameter d. The result of the proposed controller is compared with the other control techniques.

2. CEMENT MILL
A schematic representation of the cement milling circuit is depicted in Fig.1. Cement milling circuit is an industrial process, which takes raw material as input and which produces cement

having the desired fineness. The raw material enters to the classifier after grinding process in the mill. The classifier separates the incoming material into two parts. The refused material i.e. the material that is not in the desired fineness is sent back to the mill for regrinding. Accepted material goes to the other stages of the production as the output of the cement milling circuit [7].

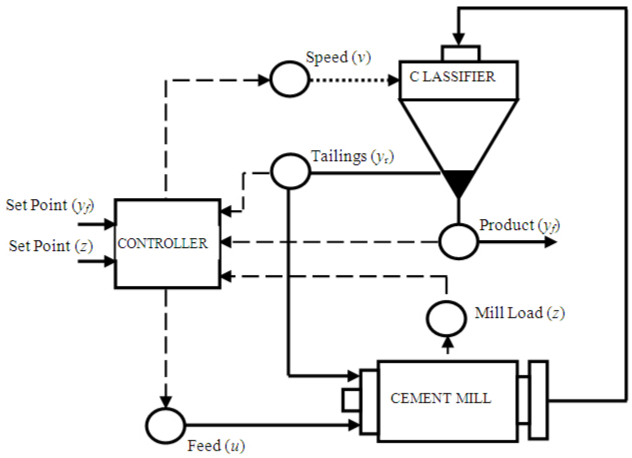

Fig. 1. Cement Milling Circuit

In steady-state operation, the product flow rate y_f is necessarily equal to the feed flow rate u while the tailings flow rate y_r and the load in the mill z may take any arbitrary constant values. The load in the mill depends on the input feed (fresh feed plus tailings flow rate) and on the output flow rate that depends in a nonlinear way, on the load in the mill and on a very important and time-varying quantity: the hardness of the material. Sometimes this nonlinearity may destabilize the system and the obstruction of the mill (a phenomenon called "plugging"), which then requires an interruption of the cement mill grinding process.

The load in the mill must be controlled at a well chosen level because too high a level of the load in the mill leads to the obstruction of the mill, while too low a circulating load contributes to fast wear of the internal equipment of the mill. Moreover, the energy consumption of the mill (i.e., the ratio energy per unit product) depends on the output of the mill that is related to the load in the mill. A usual approach is to control the tailings flow rate by using the feed flow rate as control input. This control strategy is, however, not fully satisfactory since it indirectly induces a loss of control of the product flow rate. A correct fineness of the product is also very important [5]. The fineness depends on the composition of the mill feed, but also on the rotational speed and on the air flow rate of the classifier. A natural control objective would therefore be to keep the fineness as close as possible to a desired value by controlling the rotational speed of the classifier [19]. The efficiency of a grinding circuit is dependent on three key conditions [5]:

1. An optimum and constant level of material in the mill
2. Constant air to material ratios for the separator material
3. A constant and optimum ratio between fresh feed

It can be seen that the designer could choose two of the three state variables independently, as the behavior of the third state variable would be determined upon the selection of other two. However, it is emphasized in [2],[5] that the choice of y_f and y_r may lead to unachievable values for $\varphi(z,d)$, and it is suggested in [5] that keeping y_f and z under control is a necessity.

2.1 Mathematical modeling

The following notations are introduced in Fig. 1. The mill is fed with cement clinker at a feeding rate u [tons/h]. The separator is driven by its rotational speed v [rpm]. The tailings are recycled at a rate y_r [tons /h] to the mill while the finished product exits the plant at a rate y_f [tons /h]. The plant is described by a simple dynamical model with three state variables (y_f, y_r, z) [3], [19], [20],[21].

$$T_f \dot{y}_f = -y_f + (1-\alpha(v))\,\varphi(z,d) \quad (1)$$

$$T_r \dot{y}_r = -y_r + (1-\alpha(v)\,\varphi(z,d)) \quad (2)$$

$$\dot{z} = -\varphi(z,d) + u + y_r \quad (3)$$

where T_f and T_r are time constants of the finished product and mill returns (tailings), z [tons] is the amount of material in the mill (also called the mill load), d represents the clinker hardness, $\alpha(v)$ is the separation function and $\varphi(z,d)$ is the ball mill outflow rate. The grinding function $\varphi(z,d)$ is shown in Fig.2 for different values of d. It is a non monotonic function of the mill load z. When z is too high, the grinding efficiency decreases and leads to the obstruction of the mill (plugging). A low value of z is also undesirable because it causes a fast wear of the balls in the mill.

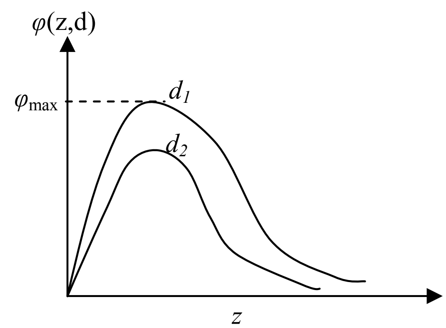

Fig. 2. Grinding function.

The separation function $\alpha(v)$, shown in Fig.3, is a monotonically increasing function of the rotational speed v of the separator, constrained between 0 and 1 with $0 \le v \le v_{\max}$ and $\alpha(v_{\max}) < 1$. The fact that with this modeling of the grinding and separation functions, the system described by (1)–(3) is positive in accordance with the physical reality:

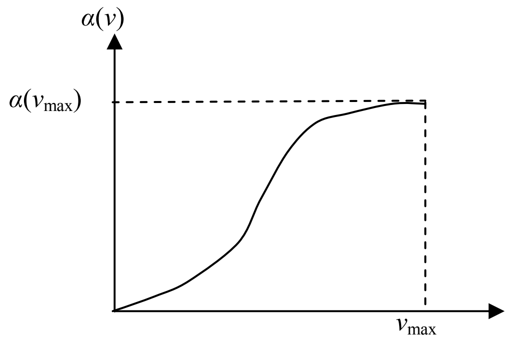

Fig.3. Separation function.

If $y_f(0) \geq 0,\ y_r(0) \geq 0,\ z(0) \geq 0,\ u_t(0) \geq 0,$

and $0 \leq v(t) \leq v_{\max}$ for all $t \geq 0$

then $y_f(t) \geq 0,\ y_r(t) \geq 0,$ and $z(t) \geq 0,$ for all $t \geq 0$.

Indeed, (1)–(3) show that whenever a component of the state becomes zero, its derivative is nonnegative.

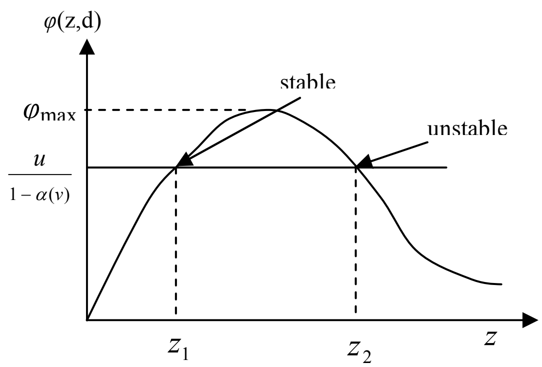

Figure 4. Equilibria and their stability

2.2 The plugging phenomenon

The grinding function is a non monotonic function of the level of material *z* in the mill, reaching a maximum for some critical value of z. When z is too high, the grinding efficiency decreases and leads to the obstruction of the mill. This nonlinearity can cause circuit instability, a phenomenon called "plugging"[19], [20], [22], [23], as shown in fig.2. The plugging phenomenon manifests itself under the form of a dramatic decrease of the production and an irreversible accumulation of material in the mill due to intermittent disturbances of the inflow rate and variations of clinker hardness [20].

In the model (1)–(3) with constant inputs $\bar{u}$ and $\bar{v}$, plugging is a global instability which occurs as soon as the state (y_f, y_r, z) enters the set Ω defined by the following inequalities as in Fig. 5 for

$y_f \geq 0,\ y_r \geq 0,\ z \geq 0,$

$$(1-\alpha(\bar{v}))\varphi(z,d) < y_f < \bar{u} \quad (4)$$

$$\alpha(\bar{v})\varphi(z,d) < y_r \quad (5)$$

$$\varphi(z,d) < 0 \quad (6)$$

$$y_f \to 0,\ y_r \to 0,\ z \to \infty \text{ as } t \to \infty \quad (7)$$

Hence, the level *z* of material in the mill is accumulated without limitation while the production rate y_f goes to zero.

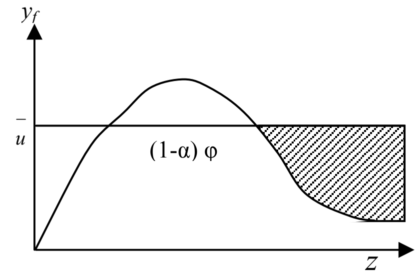

Figure 5. The plugging set Ω.

3. GENETIC ALGORITHM

Genetic algorithms (GA) are used as one of the optimization techniques. It has been shown that GA also can perform well with multimodal functions (i.e., functions which have multiple local optima). Genetic algorithms work with a set of artificial elements (binary strings, e.g., 0101010101), called a population. An individual (string) is referred to as a chromosome, and a single bit in the string is called a gene. A new population (called offspring) is generated by the application of genetic operators to the chromosomes in the old population (called parents). Each iteration of the genetic operation is referred to as a generation. A fitness function, specifically, the function to be maximized, is used to evaluate the fitness of an individual. One of the important purposes of the GA is to reserve the better schemata, i.e. the patterns of certain genes, so that the offspring may have better fitness than their parents. Consequently, the value of the fitness function increases from generation to generation. In most genetic algorithms, mutation is a random-work mechanism to avoid the problem of being trapped in a local optimum. Theoretically, a global optimal solution can be found using GA [24]. The basic operations of a simple genetic algorithm, i.e. reproduction, crossover and mutation, are described below.

3.1 Chromosome representation

Each individual coded as a binary string in the population is called a string or chromosome. The reason binary strings are preferred method of GA encoding is that information is codes as "broadly" as possible-in contrast to "compact" real numbers. The breadth, hence the resolution, of the encoding determine a Gas capability to both broadly explore and locally exploit parameter search spaces.

3.2 Fitness function

A fitness function (or objective function) is used to determine the fitness of each candidate solution. A fitness value is assigned to each individual in the population. Integral of absolute error is a better all-round performance indicator of closed loop response where overshoot, settling and rise times are the main performance considerations [14]. The IAE was therefore used as a measure of performance.

$$IAE = \int_0^t |e(t)|\, dt \quad (8)$$

In Controller Design problems IAE has to minimized, since our problem is having two control variables and the controller has to be effective in the full operating region so that weighted objective function is applied based on different setpoints of mill level (*z*) and product output(y_f) hence the objective function J is set as mentioned in equation (9)

$$J = (IAEy_{f120} \times IAEy_{f140}) + (IAEz_{60} \times IAEz_{80}) \qquad (9)$$

3.3 Selection

The selection process is centered upon the specified cost function. The selection scheme is used to draw chromosomes from the evaluated population into the next generation. Tournament selection is one of many methods of selection in genetic algorithms. Tournament selection involves running several "tournaments" among a few individuals chosen at random from the population. The winner of each tournament (the one with the best fitness) is selected for crossover. Selection pressure is easily adjusted by changing the tournament size. If the tournament size is larger, weak individuals have a smaller chance to be selected [25].

3.4 Crossover

Crossover provides a mechanism for individual strings to exchange information via a probabilistic process. Once the reproduction operator is applied, the members in the mating pool are allowed to mate with one another. First, the genetic codes of the two parents are mixed by exchanging the bits of codes following the crossover point. For example, consider two parent strings where the crossover point is 5 (i.e., the fifth bit in the string)
P1 = 10101|010; P2 = 01111|100;
The separator symbol ‘‘|” indicates the crossover site.
The resulting offspring have the following:
P01 = 10101|100; P02 = 10101|010;

3.5 Mutation

In each iteration, every gene is subject to a random change, with the probability of the pre-assigned mutation rate. In the case of binary-coding, the mutation operator changes a bit from 0 to 1, or vice versa. All in all, the mutation operation introduces new genes into the population, so as to avoid the problem of being trapped in local optima. Offspring are generated from the parents until the size of the new population is equal to that of the old population. This evolutionary procedure continues until the fitness reaches the desired specifications.

4. FUZZY LOGIC CONTROLLER

The implementation of the fuzzy logic based term is $u(t) = F[e(t), de(t)]$. In the description standard terminology is used to form fuzzy set theory, for a treatment of fuzzy sets, $e(t)$, and de(t) as inputs to the map F, and u(t) as the output. Associated with the map, F is a collection of linguistic values L={ NB, NS, ZO, PS, PB} that represent the term set for the input and output variables of F. In this case seven linguistic values are used. The meaning of each linguistic value in the term set L should be clear from its mnemonic; for example, NB stands for negative big, NS for negative small, ZO for zero and likewise for the positive (P) linguistic value. Associated with the term set L is a collection of membership functions.

$\mu = \{ \mu_{NB}, \mu_{NS}, \mu_{ZO}, \mu_{PS}, \mu_{PB} \}$

Each membership function (MF) is a map from the real line to the interval [-1 +1]. In this application the MF used is the (triangular or trapezoidal type). The height of the MF in this case is one, which occurs at the points optimized by GA. The realization of the function F[$e(t)$, $de(t)$] deals with the setting of linguistic values. This consists of scaling the inputs $e(t)$ and $de(t)$ appropriately and then converting them into fuzzy sets. The symbol C_e is the scaling constant for the input $e(t)$ and the symbol C_{de} is the scaling constant for the input $de(t)$. For each linguistic value $l \in$ L, assign a pair of numbers $ne(l)$ and $de(l)$ to the inputs $e(t)$ and $de(t)$ with the associated membership function {ne(l) = μl (C_e $e(t)$), nde(l) = μl (C_{de} $de(t)$) }. The numbers $ne(l)$ and nde(l), $l \in$ L are used in the computation of F[$e(t)$, $de(t)$] [26].

As soon as fuzzy inference is applied to each rule, the activation level for all output variable (MFs) are obtained, and the defuzzification procedure takes place. In order to compute the final control action, u(t), the most commonly used method is the center of area[26]. The result is the center of area of the profile described by the membership functions, limited in the respective activation level. Equation (10) shows the defuzzified output.

$$u^* = \frac{\int \mu_c(u) \cdot u \, du}{\int \mu_c(u) \, du} \qquad (10)$$

Where u^* is the defuzzified value,
and $\int$ denotes an algebraic integration

5. ENCODING FUZZY LOGIC CONTROL

Although fuzzy logic allows the creation of simple control algorithms, the tuning of the fuzzy controller for a particular application is a difficult task and one needs a more sophisticated procedure than that used for a conventional controller. This is due to the large number of parameters that are used to define the MFs and the inference mechanisms. Several methods have been developed for tuning fuzzy controllers. These involve adjustment of the MF [27] and scaling factors [28] and dynamically changing the defuzzification Procedure. Therefore, the approach needs as many variables as there are rules to get an optimal rule base. The advantage of the approach proposed in this paper is that it takes only three variables to optimize the rule base geometry, two variables to optimize the membership function and three scaling variables.

5.1 Encoding Rule Base

To design an optimal rule base a simple geometric approach is followed to modify the rule base as mentioned in [29] the initial assumptions are as follows;

- The magnitude of the output control action is consistent with the magnitude of the input values. (i.e. in general, extreme input values (premise) result in extreme output values (consequent), mid-range input values in mid-range output values and small/zero input values in small/zero output values.

- If a large negative (positive) input generates a large negative (positive) response, then it is likely that slightly smaller, negative (positive) inputs will necessitate a response of like polarity, but smaller magnitude, and so forth until a zero-crossover point is reached at which point the polarity of the response changes.

Using these generalizations, in conjunction with the concept of system symmetry, a different approach can be used which reduces the number of bits required for the rule -base dramatically. The approach is a variation of the method which involves a fixed coordinate system defined by the possible premise combinations. The consequent space is then 'overlayed' upon the premise coordinate system and is in effect partitioned into 5 regions shown in fig.6, where each region represents a consequent fuzzy set. The rule -base is then extracted by determining the consequent region in which each premise combination point lies. Different possible consequent space partitions are defined using 3 parameters(**C_A, C_S, CO)**

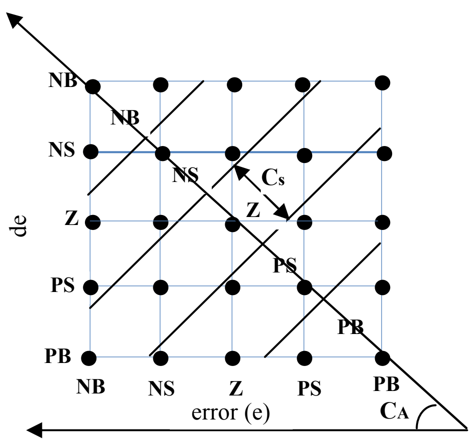

Figure 6. GAFLC Rule Assigning

Consequent-line angle, $\mathbf{C_A}$ (16 angles between 0-168^o (i.e. 4 bits))
Consequent-region spacing, $\mathbf{C_S}$ (4-bits) (C_S is a proportion of the fixed-distance between premises on the coordinate system (Ps) and is used to define the distance between consequent points along the consequent line defined by angle, CA . Its value was set to a range between (0.5 – 1.5) times the fixed distance, Ps ,using a precision of 4 bits).
Consequent-line order, $\mathbf{C_O}$ (1-bit) (Defines order of consequent space partitions (i.e. NB-NS-Z-PS-PB or PB-PS-Z-NSNB) (1 bit)
A total of 9-bits are used to extract rule -bases consistent with the above assumptions

5.2 Encoding membership function

In the attempt to encode the FLC membership functions associated with the 2 inputs and 1 output, a number of assumptions are made in respect of the distribution of fuzzy sets across the universe of discourse (UOD) for each fuzzy variable. These assumptions are;

- The UOD is symmetrical about the central, zero region for each variable.
- The extreme membership functions (MF) for input variables should be unbounded in the respective positive and negative going directions.
- The inner and central UOD-range MFs could assume either triangular (trimf) or trapezoidal (trapmf) shapes only, for input and output variables. Outer UOD-range MFs for input variables were unbounded z-shaped (zmf), while output variable extreme MFs could assume the same shape as inner and central range MFs (trimf or trapmf).
- The number of fuzzy sets for the controller was fixed at 5 (NB, NS, Z, PS PB).

The MF properties altered by the GA are as follows;
MF shape (triangular or trapezoidal). Degree of MF-centre shift to effect MF compression or expansion.
All evaluated FLCs contain 3 variables, e (error), de (error-derivative) and u (control-action). For the input variables, e and de, and output variable, u, 7 bits are used to define the properties of the MFs to be optimized. For each variable, their respective 7-bit GA-chromosome segments are sub-divided into 2 fields;

1. The "offset field" (3 bits) used to effect change of shape of the MFs.
2. The "companding factor" field (4 bits) used to effect expansion/compression of the MFs.

5.3 MF Offset Field

The optimization begins by loading a *.fis (Matlab Fuzzy file) into the FLC block in the MATLAB Simulink model. Each evaluation subsequently uses a 'genetically-altered' version of the original FLC which is defined by a MATLAB, fuzzy structure. For each evaluated FLC, the UOD -distributed MFs are initially assumed to be trapezoidal in type, thus 4 parameters are required by the FIS to define the position in the UOD of each of the 5 MFs. The significance of these parameters is illustrated below in Figure7 & 8. The Matlab Fuzzy file 'params' field has 4 UOD position parameters (outer-left(OL), inner-left(IL), inner-right(IR), outer-right(OR)). For inner parameters (IL and IR) equal in value, MF becomes triangular in shape. The offset field is used to effect a change of shape in the MFs.

The 3-bit offset field is decoded in the range of [0, 0.1] and the application of the offset parameter modifies the shape of the MFs from triangular to trapezoidal of varying widths and positions. The MFs of each FLC fuzzy variable (e, de and u) are encoded into the GA-chromosome in this manner.

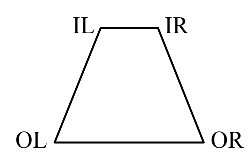

Fig.7 Trapezoidal MF parameters

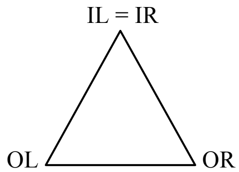

Fig 8. Trapezoidal MF Defining triangular MF

5.4 MF Companding Field

Application of the offset field produces MFs of different shapes (trimf or trapmf) and positions, but does not effect the distribution of the MFs, which are evenly distributed across the UOD. To enable evaluation of non-uniform distributed MFs, a further field is encoded into the GA-chromosome for each fuzzy variable, which is applied to the MFs to bring about compression and/or expansion of the associated MFs. The companding field is decoded to a value (CF) in the range [0.5 – 2], and is applied to update the MF position parameters of each MF by raising them to the power of CF (e.g. for the Z-MF, outer-left parameter; $o_{OL(new)} \Rightarrow (o_{OL(old)})^{CF}$) Due to the use of a normalized UOD, the position parameters are shifted to different degrees by this operation and the net effect is that;

for CF < 1 : Z-MF is compressed, NB and NS expand
for CF > 1 : Z-MF expands, NB and NS compress
for CF = 1 : uniform MF distribution
In this way, non-uniform distribution of the MFs is effected across the UOD. As is the case in relation to MF shapes, companding fields for each fuzzy variable (e, de and u) are encoded into the GA-chromosome.

5.5 Encoding FLC Scaling Gains

The GA also attempts to optimize the scaling gains of the *e* and *de* inputs of the fuzzy controller. Three fields, e-scaling (C_e), de-scaling(C_{de}) and output C_u are included in the GA chromosome each consisting of 7- bits, which are encoded to yield values of gain for the appropriate gain blocks of the Simulink model used to evaluate each controller.

5.6 GA-Chromosome of FLC

Three aspects of the FLC were subject to the optimization procedure; (a) Rule Base, (b)Membership Functions(MF), (c) Input Output Scaling Gains. The primary assumption made was that for a symmetrical system, a corresponding FLC would also exhibit symmetry about the set point in respect of its MFs and rule -base. This assumption was exploited in order to attempt to reduce the number of bits required to define the FLC for GA optimization. Figure 9 illustrates the 102-bit binary GA-chromosome used to encode two FLC output

6. CONTROL STRUCTURE

The control structure is shown in the fig.10, Based on the principles of the Expert control algorithm, shown in figure11 the MIMO fuzzy logic controller is optimized using GA for a cement mill process. The control objective is to regulate the finished product rate y_f and the mill load z at the desired set points y_f and z by manipulating the feed flow rate u and the separator speed v. for the ideal operating conditions the setpoint of mill level z is set as 65 tons and finished product y_f is set as120 (tons/h) and the material hardness *(d)* is varied from 1 to 1.4. Four inputs is given to the fuzzy logic controller is the mill level error *(e_z)*, mill level error rate *(de_z)*, finished product error *(e_{yf})* and finished product error rate *(de_{yf})*.

7. SIMULATION

The effectiveness of the proposed control law has been assessed through simulations where the model (1)–(3) represents the plant with analytical forms for the φ and α functions which satisfy the shape assumptions given in Sections 2

$$\varphi(z,d) = 20z\,\exp\left(-\frac{dz}{80}\right)$$

$$\alpha(v) = 9\left(-\frac{v}{v_{\max}}\right)^3 - 13.5\left(-\frac{v}{v_{\max}}\right)^4 + 5.4\left(-\frac{v}{v_{\max}}\right)^5$$

$v_{\max} = 200,\ \alpha_{\max} = 0.9$ And the time constants T_f = 0.3 [h], T_r = 0.01 [h]. These functions have been tuned in order to match experimental step responses of an industrial cement grinding circuit[31]. The entire simulation is carried out in MATLAB & Simulink on a Core 2 Duo Processor 2.2 GHz, 2GB RAM PC Environment.

7.1 Case I

For optimizing the Fuzzy logic controller, the GA parameters are set to:

Generation	=	250
Population Size	=	50
Crossover rate	=	0.5
Mutation Rate	=	0.03

The figure 12 shows the GA optimized MFs of the FLC

FLC Chromosome for Mill Feed control (u)							FLC Chromosome for Classifier Speed control (v)						
RB 9bits 1:9	e:MF 7 bits 10:16	de:MF 7 bits 17:23	u:MF 7 bits 24:30	e:scal 7 bits 31:37	de:scal 7 bits 38:44	u:scal 7 bits 45:51	RB 9bits 52:60	e:MF 7 bits 61:78	de:MF 7 bits 79:84	u:MF 7 bits 85:82	e:scal 7 bits 83:90	de:scal 7 bits 91:96	u:scal 7 bits 97:102

Figure 9. GAFLC-chromosome structure

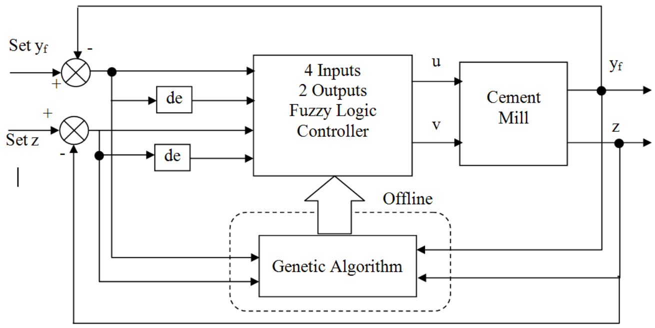

Figure.10 Control structure of GAFLC for cement mill

7.2 Case II

For testing the GA optimized Fuzzy logic controller, the following settings are chosen, with Initial setpoint values: y_f = 120 tons/h and z = 60 tons.

$$Set\ y_f(t) = \begin{Bmatrix} 120 & for\ 0 < t < 3 \\ 140 & for\ 3 < t < 11 \end{Bmatrix}$$

$$Set\ z(t) = \begin{Bmatrix} 60 & for\ 0 < t < 6 \\ 70 & for\ 6 < t < 11 \end{Bmatrix}$$

$$d(t) = \begin{Bmatrix} 1 & for\ 0 < t < 8 \\ 1.40 & for\ 8 < t < 11 \end{Bmatrix}$$

The set-point for the product flow rate y_f is changed from 120 to 140 tons/h at time t=3 hours and the set-point for the mill level z is changed from 60 to 70 tons at time t=6 hour, the hardness d varied from its nominal value 1 to 1.5 at time t=8. The closed loop response of the Cement mill for the following settings is shown in fig.13.

7.3 Case III

To check the disturbance rejection of the GAFLC, the hardness parameter (d) is varied from 1.34 to 1.8 the setpoints of y_f is set as 120 tons/h and z is set as 60 tons. The hardness change is introduced at time t=6 hour and the response is plotted for the different hardness values (d=1.34,1.40,1.45,1.50,1.60,1.70,1.80). The response for y_f and z are shown in fig.15.

8. RESULTS AND DISCUSSIONS

The Table 1 shows the optimized fuzzy logic control variables. It is observed that the performance Index is minimized to 403.2 from 2957.3 after 250 generations and the fig.12 shows the optimized membership functions after 250 generations.

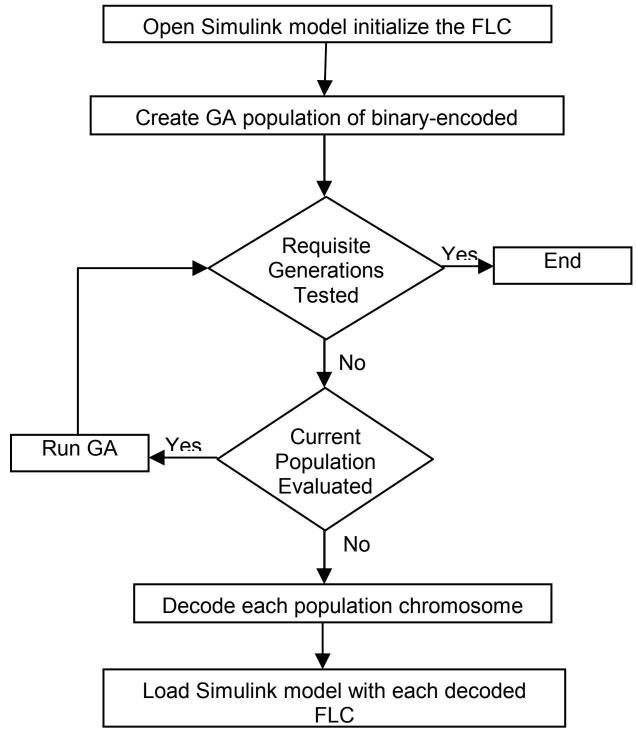

Figure 11. GAFLC Design flow chart

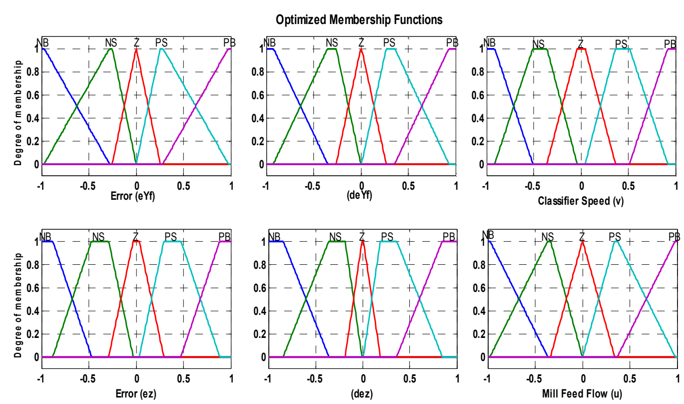

Figure 12. GA optimized FLC membership functions

Table 1 Optimized fuzzy logic control variables

	Rule base parameters			Membership Parameters						Scaling parameters			J
	Ca	Cs	Co	Ofset1	Ofset2	Ofset3	Cf1	Cf2	Cf3	Ke	Kde	Kop	
FLC u 1st Gen	1.77	0.06	1	0.06	0.04	0.03	0.50	0.80	0.70	1.07	1.02	11.67	2957.3
FLC z 1st Gen	2.95	1.50	0	0.01	0.03	0.07	1.00	0.50	0.60	1.09	0.33	6.27	
FLC u 250thGen	0.78	.96	1	0.07	0.08	0.03	0.9	1.25	0.58	0.98	1.41	23.23	403.2
FLC z 250thGen	0.59	1.30	1	0.09	0.06	0.04	1.30	1.00	0.70	1.11	0.22	15.48	

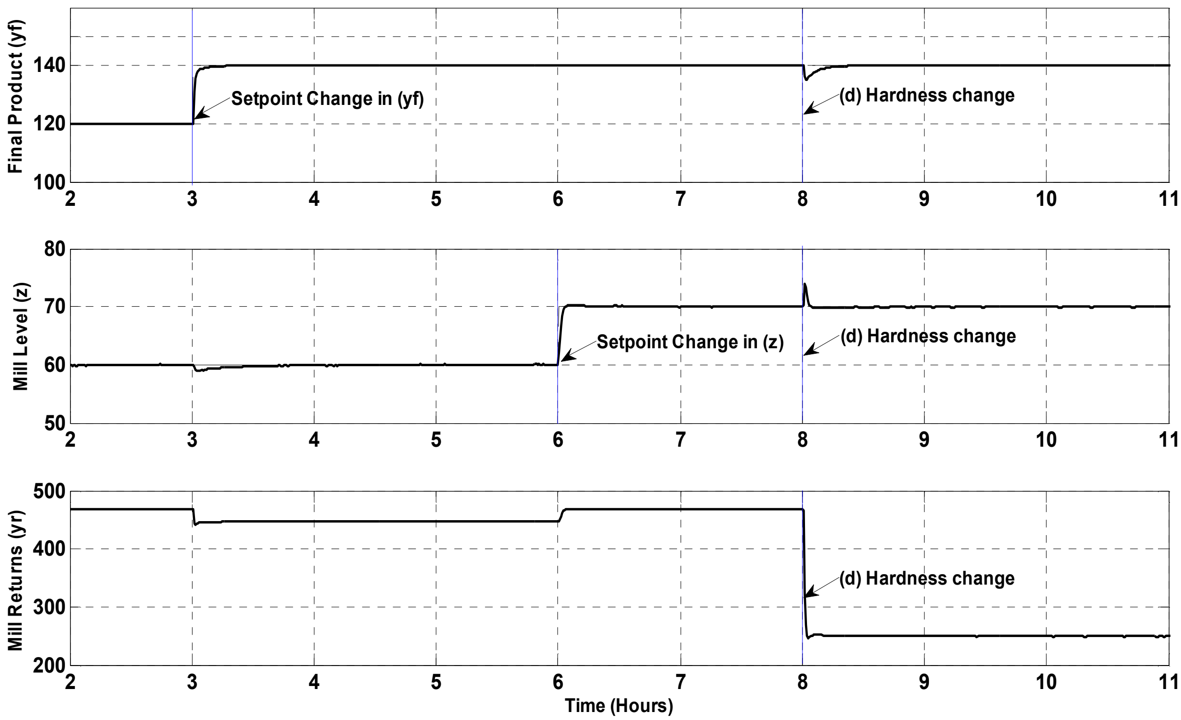

Figure 13. Closed loop response of GAFLC for varying setpoints and hardness

The fig.13 shows the closed loop response of the simulated cement mill circuit with GA optimized Fuzzy logic Controller for the set point profile and the hardness profile given in section 7.2, three variables (y_f, z, and y_r) are plotted for 11 hours time when there is a sudden rise in product outflow setpoint (y_f) , the controller is capable of keeping the controlled variable in the set value with out overshoot and with quick settling time, similarly for the mill level (z) the controller is very effective. It is noted that when there is a change of setpoint for (y_f) or (z) there is only a small deviation in the other loop compared to the other control strategy reported in the literature (Nonlinear robust controller [20], Nonlinear receding horizon (NRH) control [30], linear quadratic control [30], Nonlinear learning control [7], Neural Network based control [21]), also the effect of hardness change does not destabilize the cement mill.

The fig.14 shows the error output for the case II simulation settings. It is noticed that the error after the hardness varied from 1 to 1.4 the controller is capable of bringing back the error of (y_f) and (z) to zero.

Table 2 represents the comparison of the closed response output for set point variations keeping hardness (d) as 1. The performance of the controllers are compared with respect to the risetime (Rt), Peak overshoot (Po), Peak undrshoot (Pu), and settling time (St). The GA optimized FLC seems to be better in all performances.

Table 3 represents the comparison of the closed response output for the change in the hardness parameter (d) from 1 to 1.4. The proposed control scheme is performs better with minimum Rt, St, Po and Pu, in where as linear quadratic control cannot stabilize the mill. By comparing the performances shown in table 2 & 3, the proposed control scheme performs better than the other control schemes.

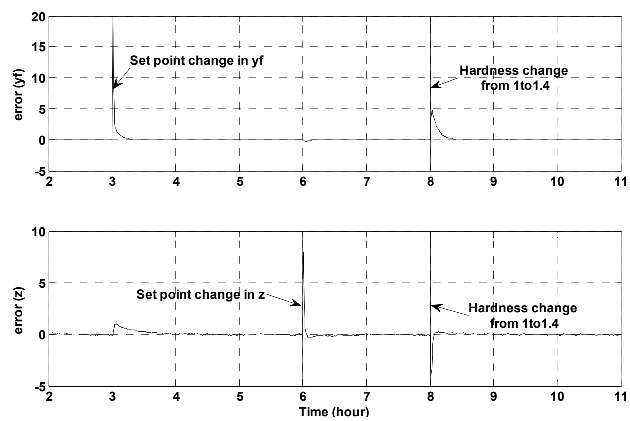

Figure 14. Error output of (yf) and (z) using GAFLC for varying setpoints and hardness

Fig.15 depicts the output response for the simulation settings ginven in Case III. The response for (y_f) and (z) is plotted for different hardness parameters (varying from 1.34 to 1.8) the response shows the settling time for all hardness is more or less same. The hardness parametter is varied upto 1.8 which was not studied in privous work. The Table 4 shows the performance measure of (y_f) and (z) for the proposed controller scheme at various hardness values.

Table 2. Performance comparison of controlled variable for varying setpoint

Controller types	Finished Product (y_f)			Mill Load (Z)		
	Raise Time (Min)	Peak overshoot (%)	Settling time (Min)	Raise Time (Min)	Peak overshoot (%)	Settling time (Min)
GAFLC (proposed)	4	0	15	5	.03	15
Nonlinear robust controller*	26	4.64	181	16.5	0	70
Nonlinear receding horizon (NRH) control*	51	1.58	260	27	4.1	240
linear quadratic control*	50	1.45	255	30	4.1	250
Neural Network based control*	4	1.13	13	8	1.7	20

Table 3. Performance comparison of controlled variable for hardness change

Controller Type	Finished Product (y_f)			Mill Load (Z)	
	Peak over shoot (%)	Under Shoot (%)	Settling time (Min)	Peak over shoot (%)	Settling time (Min)
GAFLC (proposed)	0	1.66	18	3.63	8
Nonlinear robust controller*	5.53	15.62	182	17.5	168
Nonlinear receding horizon (NRH) control*	0.8	13.33	410	19.6	210
linear quadratic control*	UNSTABLE				
Nonlinear learning control*	0	6.66	42	3.70	35
Neural Network based control*	0.3	1.25	18	3.13	12

*Data taken from the response of [7],[20],[21],[31].

Table 4. Performance measure of GAFLC controller for different hardness (d)

Hardness Value (d)	Finished Product (y_f)		Mill Level (z)	
	IAE	Undershoo t (%)	IAE	Overshoot (%)
1.34	2.71	1.66	0.94	3.63
1.40	3.57	2.16	1.08	4.31
1.45	3.59	2.66	1.13	5.00
1.50	4.60	3.25	1.51	5.66
1.60	6.10	4.30	1.89	7.16
1.70	7.98	5.91	2.38	8.66
1.80	10.46	8.41	2.8	10.03

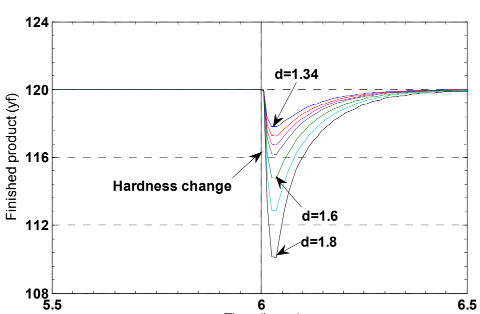

Figure 15. Output response of finished product (yf) for different hardness parameter (d)

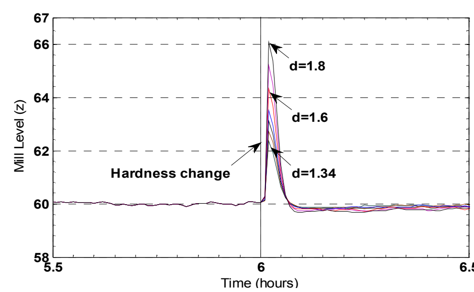

Figure 16. Output response of mill level (z) for different hardness parameter (d)

9. CONCLUSION

Optimization of a fuzzy logic controller can prove a lengthy process when performed heuristically. In this work it has been shown that the use of genetic algorithms offers a feasible method for the optimization of the knowledge-base of fuzzy logic controllers. The literature contends that optimization of a FLC can be considered as a geometric search problem of a multimodal hyper surface. The proposed approach shows a good performance in building the fuzzy logic controllers for a complex Cement mill process. The performance of our fuzzy controller is tested with cement mill circuit via simulation, and the results are compared with other control techiniques proposed in [7],[20],[21] and [30]. The results demonstrate that the FLC controller designed by the proposed method is robust and efficiently control a complex Cement mill process from plugging.

REFERENCES

[1] Gajjar, J.T, Rao, A.V.M, Wolfinger, J.F, "PC-based diagnostic tool for predictive maintenance of cement mill-equipment",Cement Industry Technical Conference, 15-17 May 1989 Page(s):215 - 223

[2] M. BoulTin. M.Boulvin, C.Renotte, A.Vande Wouwer, Remy, and R. Lepore, "Modeling, simulation and evaluation of control loops for a cement grinding process", IEEE Transactions on Control Systems Technology 11 (2003) 715 - 725

[3] Jadot, F.; Bastin, G.; Wertz, V.; Magni, L., "Global state feedback stabilisation of cement mills", 1998. Proceedings of the 37th IEEE Conference on Decision and Control,Volume 3, Issue , 1998 Page(s):3168 - 3170

[4] C. Cigaieck and K. Krevsa "Two-paranmeter control svstenm for a cement grinding process." Translation of Zement-Kalkc-Gips. 1991. Pages 202-206.

[5] V.Van Breusegen, L.Chen, G.Bastin and V.Wertz, V.Werbrouck, C. de Pierpont "An Industrial Application of Multivariable Linear Quadratic Control to a Cement Mill Circuit", IEEE Transactions on Industry Application, vol.32 pp.670-677, 1996.

[6] de Haas, B.; Werbrouck, V.; Bastin, G.; Wertz, V. "Cement mill optimization: design parameters selection of the LQGcontroller", Proceedings of the 4th IEEE Conference on Control Applications 28-29 Sep 1995 Page(s):862 – 867

[7] Dagci, O.H. Efe, M.O. Kaynak, O, "A nonlinear learning control approach for a cement milling process", 2001. (CCA '01). Proceedings of the 2001 IEEE International Conference on Control Applications page(s): 498-503

[8] Andon V. Topalov, Okyay Kaynak, "Neuro-adaptive modeling and control of a cement mill using a sliding mode learning mechanism", IEEE International Symposium on Industrial Electronics,May 4-7 2004, pages. 225-230,

[9] Lee, C. C. "Fuzzy logic control systems: fuzzy logic controller part-I" IEEE Transactions on Systems, Man and Cybernetics, (1990). 20(2), 404–418.

[10] A.M.F. Fileti, A.J.B. Antunes, F.V. Silva, V. Silveira Jr, J.A.F.R. Pereira Experimental investigations on fuzzy logic for process control"Control Engineering Practice, Volume 15, Issue 9, September 2007, Pages 1149-1160.

[11] Sahasrabudhe, R.; Sistu, P.; Sardar, G.; Gopinath, R. "Control and optimization in cement plants", Control Systems Magazine, IEEE Volume 26, Issue 6, Dec. 2006 Page(s):56 – 63

[12] Dragan.D.K,. Kuzmanovic.S.B,. Emil.L *"Design of a PID-like compound fuzzy logic controller"*Engineering Applications of Artificial Intelligence, 14 (2001) 785-803

[13] Shi.Yuhui.,. Eberhart.Russell,. Chen.Yaobin *"Imple-mentation of Evolutionary Fuzzy Systems"* IEEE Transactions on Fuzzy Systems, Vol. 7, No. 2, April 1999

[14] Visioli.A *"Tuning of PID controllers with fuzzy logic"* IEEE Proc.-Control Theory Appl., Vol.148, No.1, January 2001

[15] Hyun-Joon.Cho,. Kwang-Bo.Cho,. Bo-Hyeun.Wang *"Fuzzy-PID hybrid control: Automatic rule generation using genetic algorithms"* Fuzzy Sets and Systems 92 (1997) 305-316

[16] Gurocak ,H.B. *"A Genetic Algorithm method for tuning Fuzzy Logic Controllers"* Fuzzy Sets and Systems 108 (1999) 305-316

[17] Chaiyaratana, N.; Zalzala, A.M.S. "Recent developments in evolutionary and genetic algorithms: theoryand applications"Genetic Algorithms in Engineering Systems: Innovations and Applications, Second International Conference On (Conf. Publ. No. 446) 2-4 Sep 1997 Page(s):270 - 277

[18] Cordon, O.; Alcala, R.; Alcala-Fdez, J.; Rojas, I. "Guest Editorial Genetic Fuzzy Systems: Whatapos;s Next? An Introduction to the Special Section", IEEE Trans. on Fuzzy Systems Volume 15, Issue 4, Aug. 2007 Page(s):533 – 535

[19] A. Ramezani, H. Ramezani, B.Moshiri, The Kalman Filter Information Fusion for Cement Mill Control Based on Local Linear Neuro-Fuzzy Model, proceedings of 4th International Conference on Innovations in Information Technology, (2007), pp. 183 –187

[20] F.Grognard, F.Jadot, L.Magni, G.Bastin, R.Sepulchre, V.Wertz, Robust stabilization of a nonlinear cement mill model, IEEE Transactions on Automatic Control 46 (2001) 618 –623

[21] Topalov A.V., Kaynak.O., "Neural network modeling and control of cement mills using a variable structure systems theory based on-line learning mechanism" Journal of Process Control 14 (2004) 581–589

[22] I.Kotini, G.Hassapis, A Hybrid Automaton Model of the Cement Mill Control**,** IEEETransactions on Control Systems Technology 16 (2008) 676 – 690

[23] R. Sahasrabudhe, P. Sistu, G. Sardar, R. Gopinath, Control and optimization in

a. cement plants, IEEE Control Systems Magazine 26 (2006) 56-63

[24] P.C. Chen, C.W. Chen and W.L. Chiang, "GA-based modified adaptive fuzzy sliding mode controller for nonlinear systems", Expert Systems with Applications 36 (2009) 5872–5879

[25] Brad L. Miller and David E. Goldberg.1995, Genetic Algorithms, Tournament Selection and the Effects of Noise, vol 9, pp 193-212

[26] Godwin Anand and Albertraj, Proceedings of International Conference on Intelligent Knowledge Systems (IKS-2004), August 16-20,(2004) paper 36

[27] Arslan, A.,Kaya,M.,2001. Determination of fuzzy logic membership functions using genetic algorithms. Fuzzy Sets and Systems118,297–306.

[28] Zhao, Y.,CollinsJr.,E.G.,2003. Fuzzy parallel parking control of autonomous ground vehicles in tights paces. In: IEEE International Symposium on Intelligent Control, Houston,TX,pp.811–816.

[29] Sk.Faruque Ali, Ananth Ramaswamy, "Optimal fuzzy logic control for MDOF structural systems using Evolutionary algorithms" Engineering Applications of Artificial Intelligence 22(2009)407–419

[30] L. Magni, G. Bastin, and V. Wertz, "Multivariable nonlinear predictive control of cement mills," IEEE Trans. Contr. Syst. Technol., vol. 7, pp. 361–371, July 199