MULTI-MODEL SYSTEM WITH NONLINEAR COMPENSATOR BLOCKS

Ciprian LUPU[1], Cătălin PETRESCU[2], Alexandru ŢICLEA[3], Cătălin DIMON[4], Andreea UDREA[5], Bogdan IRIMIA[6]

Structurile de reglare multimodel sunt o soluție viabilă pentru conducerea unor sisteme cu neliniarități importante sau cu regimuri de funcționare multiple. Una din problemele specifice structurii este determinarea numărului de modele. Cu cât numărul de modele este mai mare, cu atât performanțele sunt mai bune, dar şi complexitatea structurii creşte. Lucrarea propune o metodologie originală pentru reducerea numărului de modele fără reducerea performanțelor sistemului.

Soluția are o valoarea practică fiind uşor de implementat pe structurile hardware bazate pe automate programabile şi calculatoare de proces. Rezultatele experimentale probează performanțele structurii.

The multi-model control structures represent real solutions to control the systems with important nonlinearities or different functioning regimes. One of structure's specific problems is determination of models number. An increased number determine superior performance and very complex structure. The paper proposes a original methodology for reducing the model number without performance decreasing.

This solution has practical importance being facile to be implemented on PLC and process computers. Experimental results prove structure's performances.

Keywords: multi-model, control, nonlinearities, compensator block, real-time

1. Introduction

The multi-model systems represent a relatively new approach on nonlinear control strategies. Since the 90's different studies for the multi-model control

[1] Reader, Department of Automatic Control and Computer Science, University POLITEHNICA of Bucharest, Romania, cip@indinf.pub.ro
[2] Reader, Department of Automatic Control and Computer Science, University POLITEHNICA of Bucharest, Romania, catalin@indinf.pub.ro
[3] Lecturer, Department of Automatic Control and Computer Science, University POLITEHNICA of Bucharest, Romania, ticleaa@ yahoo.com
[4] Assistant, Department of Automatic Control and Computer Science, University POLITEHNICA of Bucharest, Romania, catalin.dimon@gmail.com
[5] Assistant, Department of Automatic Control and Computer Science, University POLITEHNICA of Bucharest, Romania, udrea.andreea@yahoo.com
[6] Assistant, Department of Automatic Control and Computer Science, University POLITEHNICA of Bucharest, Romania, irimiab@zappmobile.ro

strategy have been developed. The Balakrishnan's and Narenda's first papers which proposed several stability and robustness methods using classical switching and tuning algorithms have to be mentioned [1]. Further research in this field determined the extension and improvement of the multi-model control concept. Magill and Lainiotis introduced the model representation through Kalman filters. In order to maintain the stability of minimum phase systems, Middelton improved the switching procedure using an algorithm with hysteresis. Petridis', Kehagias' and Toscano's work focused on nonlinear systems with time variable. Landau and Karimi have important contributions regarding the use of several particular parameter adaptation procedures, namely CLOE (Closed Loop Output Error) [2]. The multi-model control version proposed by Narenda is based on neural networks [1]. Finally, Dubois, Dieulot and Borne apply fuzzy procedures for switching and sliding mode control.

The structure of a multi-model control system depends on process particularities.

In this paper is proposed a multi-model control structure which contains for each model/controller a nonlinearity compensator. This solution allows a reduced number of models and a reduced complexity for global structure. Its base consists on determination of static characteristic for each model.

This structure can be applied in the case of processes with important nonlinearities.

2. Classic solution for multi-model structure

The classic solution consists on a set of models:

$$M = \left\{ \mathcal{M}_1, \mathcal{M}_2, \mathcal{M}_3 \ldots \mathcal{M}_n \right\},$$

and on a set of the correspondent controllers:

$$C = \left\{ C_1, C_2, C_3 \ldots C_n \right\},$$

integrated in the closed-loop configuration as presented in Figure 1.

The input and output of the process P are u and y respectively, and r is the set point of the system. The M_i (i=1,2,... n) models are *a priori* evaluated. For each model M_i a controller C_i is designed in order to assure the nominal performances for the pair (M_i, C_i).

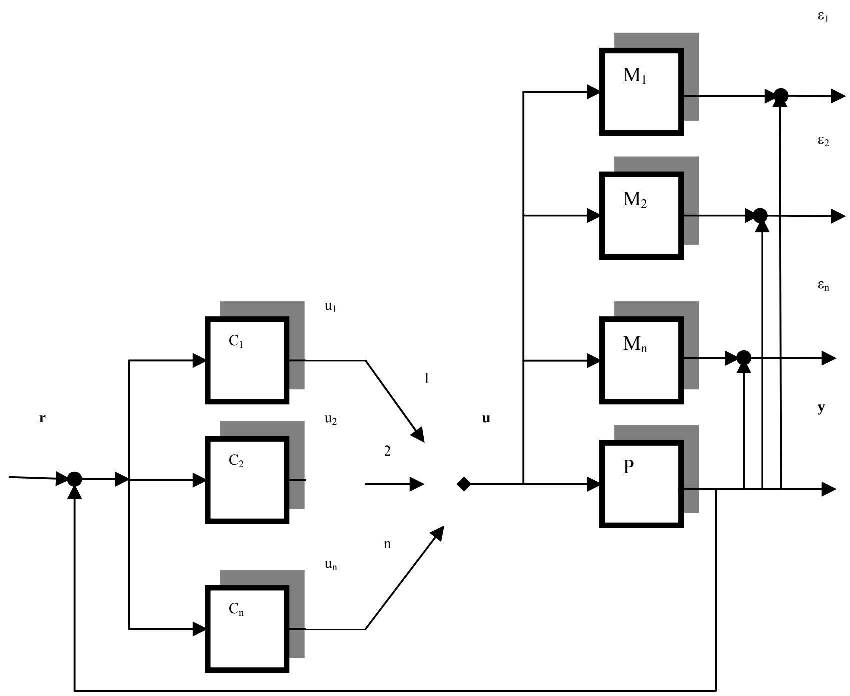

Fig. 1. Multi-model control scheme.

The main idea of the multi-model structure construction consists on dividing of process functioning zone in an ***n*** number of small zones where the models are more simplified and where the ***n*** corresponding control algorithms has decreased complexities (Figure 2).

One of the principles used in zone's choosing is that the difference between "linear" characteristic and real characteristic has a lower value, under imposed value. In fact, a small zone does not impose a corresponding identified linear model. Is very possible to have a second or third or *m* order model, but this implies a complex corresponding control algorithm. A very complex algorithm can determine a superior performances and important hardware resources on real time implementation.

Real situation imposes a balance between complex model/algorithm and complex real time hardware/software architectures.

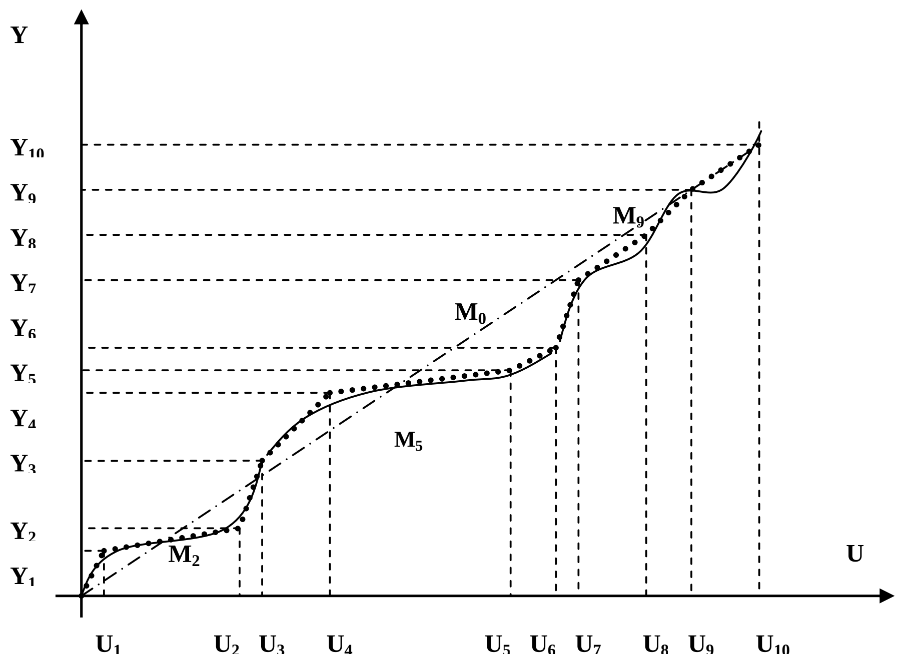

Fig. 2. Construction of the set of process's models

In Figure 2 continuous line represents real static characteristic, dot line "linear" models and dash and dot line is "global linear" model. The last one has an important distance from real characteristic (maximal in U_6, Y_6 point). A single controller has low performance for "normal" complexities. If high performance is desired, one needs a complex algorithm, for a robust implementation.

3. Proposed solution for multi-model structure

In this paper is proposed a multi-model control structure which contains for each controller a nonlinearity compensator [3]. This solution allows a reduced number of models and a reduced complexity for each algorithm. Its base consists on determination of static characteristic for each model. This solution is also named "control system with inverse model".

In the literature there are a lot of inverse model proposed structures. According to them, the paper selects a very simple and efficient structure presented in Figure 3. This solution supposes adding of two commands: the first one "a direct command" generated by inverse model command generator, and the second generated by a classic and very simple algorithm (PID, RST etc.).

This structure is multiplied in all contained controllers in multi-model structure. So, for each controller, the first command, based on process static characteristics, is dependent on set point value and is designed to generate a corresponding value to drive the process's output close to imposed set point value. The second (classic feedback) algorithm generate a command that, correct the difference caused by external disturbances and according to set point, by eventual bias error caused by mismatches between calculated inverse process characteristic and situation from real process.

Presented solution proposes treating of these inverse model mismatches that "disturb" the first command as a second command classic algorithm's model mismatches. This solution imposes designed of classic algorithm with a corresponding robustness reserve. For this reason designing of the second algorithm is made in two steps:

- designing of a classic algorithm base on a model identified in a real functioning point – selected fortuity or, on the middle of corresponding segment process characteristic;
- verification of algorithm's robustness and improving of this, if it is necessary in a new (re)designing procedure;

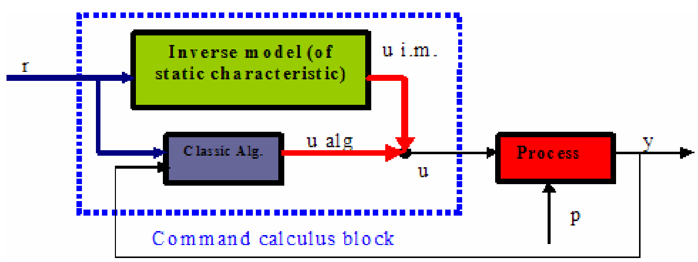

Fig. 3. Proposed scheme for inverse model structure

On figure 3, the blocks and variables are as follows:

- Process – physical system to be controlled;
- Command calculus – unit that computes the process control law;
- Classic Alg. – control algorithm (PID, RST);
- y – output of the process;
- u – output of the Command calculus block;
- u alg. – output of the classic algorithm;
- u i.m. – output of the inverse model block;
- r – system's set point or reference trajectory;
- p – disturbances of physical process.

This solution used in context of multi-model structure has three important aspects:

- Selection of a reduced number of "major zones" where the nonlinearity is important but lower than an imposed value.

- Construction of compensator block for each “major zone”.
- Designing of correspondent controller for each “major zone”.

All three will be presented in next sections.

3.1 Selection of “major zones”

How are selected the “major zones”? The number of “major zones” must by much reduced (2, 3 or maximum 4) and these can consist in medium or “local” tendencies of nonlinear characteristic [4], [5]. Figure 4 presents an example for this aspect. It can be imposed that the difference between the tendency and real characteristic is less or equal to an imposed margin.

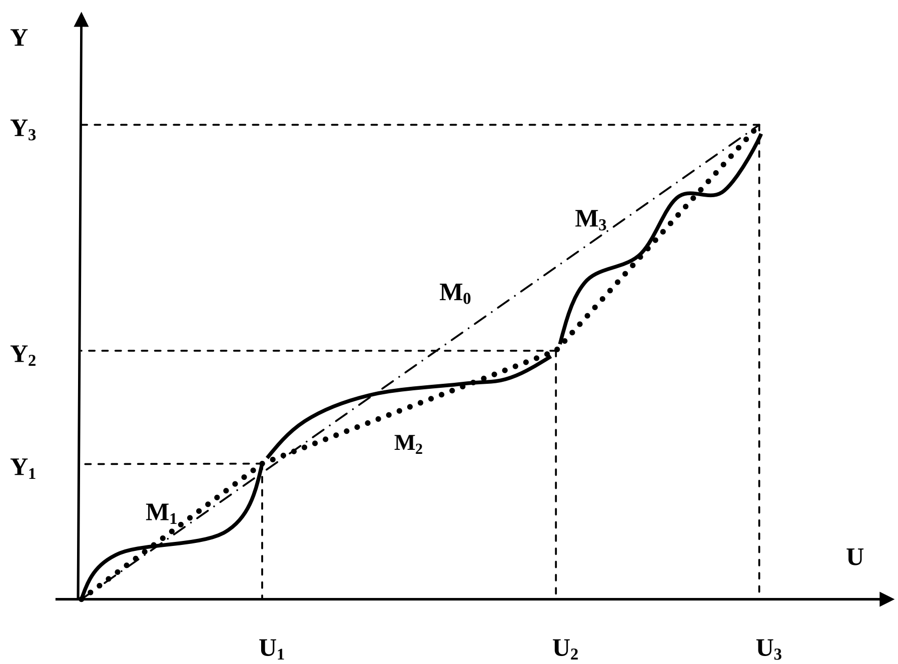

Fig. 4. Selection of major zones

In Figure 4 continuous line represent real static characteristic, dot line “linear” models and dash and dot line is “global linear” model.

3.2 Construction of nonlinear compensator blocks

This operation is based on several experiments of discrete step increasing and decreasing of the command *u(k)* and measuring the corresponding stabilized process output *y(k)*. The command *u(k)* cover all possibilities (0 to 100% in percentage representation). Because the process is disturbed by noises, usually the static characteristics are not identically. The final static characteristic is obtained by meaning of correspondent position of these experiments. Figure 5 presents this operation. The graphic between two “mean” points can be obtained using extrapolation procedure.

According to system identification theory [4] the dispersion of process trajectory can be finding using expression (1):

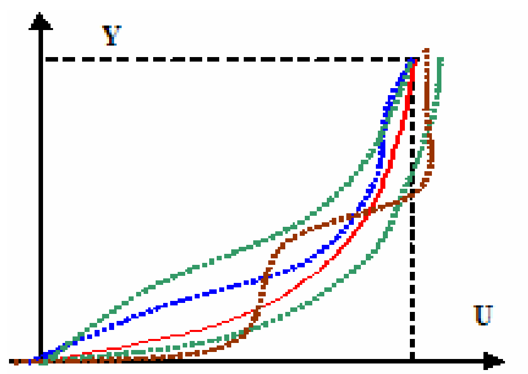

Fig. 5. Determination of static characteristic of the process.
Red (continuous) line represents the final characteristic.

$$\sigma^2[n] \cong \frac{1}{n-1}\sum_{i=1}^{n} y^2[i],\ \forall \mathrm{n} \in N^* \setminus \{1\} \tag{1}$$

This can express a measure of superposing of noise that action onto process, process’s nonlinearity etc. and is very important on control algorithm designed robustness.

Other possibility is to find the position and the value of the maximal distance (noted – *md)* from “mean” characteristic.

Next step for nonlinear compensator block deals with the „transposition” operation of the process’s static characteristic. Figure 6 presents this construction. According to this, *u(k)* is dependent to *r(k)*. This characteristic is stored in a table; thus we can conclude with this, for the inverse model based controller, that by selecting a new set point *r(k)* it will be necessary to find in this table the corresponding command *u(k)* that determines a process output *y(k)* close to the reference value.

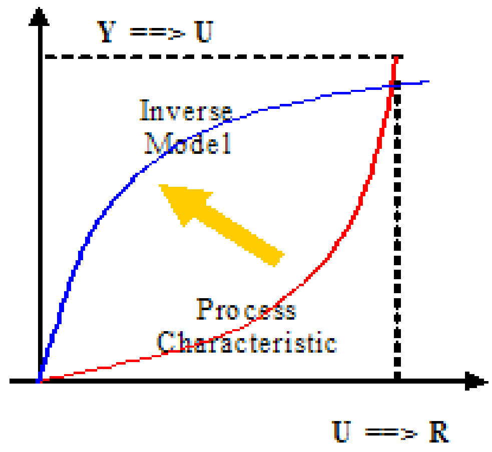

Fig. 6. Construction of inverse model

3.3 Designing of controllers

For the whole "major" zone, control algorithm's duty is to eliminate the disturbances and differences between inverse model computed command and real process behavior. A large variety of control algorithms can be used here, PID, RST, fuzzy etc., but the goal is to have a very simplified one.

For this study we use a RST algorithm. This is designed using pole placement procedure [6]. Figure 7 presents a RST algorithm:

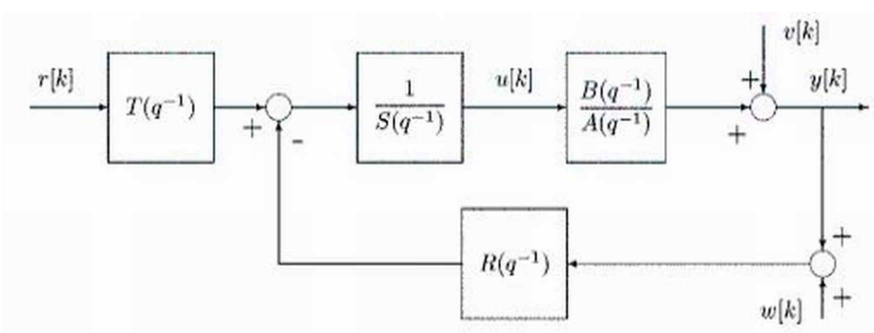

Fig. 7. RST control algorithm structure

Where R, S, T polynoms are:

$$R\left(q^{-1}\right)=r_0+r_1q^{-1}+\ldots+r_{nr}q^{-nr}$$
$$S\left(q^{-1}\right)=s_0+s_1q^{-1}+\ldots+s_{ns}q^{-ns} \quad (2)$$
$$T\left(q^{-1}\right)=t_0+t_1q^{-1}+\ldots+t_{nt}q^{-nt}$$

Algorithm pole placement design procedure is based on identified process's model.

$$y(k)=\frac{q^{-d}B(q^{-1})}{A(q^{-1})}u(k) \quad (3)$$

where

$$B\left(q^{-1}\right)=b_1q^{-1}+b_2q^{-2}+\ldots+b_{nb}q^{-nb}$$
$$A\left(q^{-1}\right)=1+a_1q^{-1}+\ldots+a_{na}q^{-na} \quad (4)$$

The identification is made in a specific process operating point and can use recursive least square algorithm exemplified in next relations developed by Landau in [6]:

$$\hat{\theta}(k+1)=\hat{\theta}(k)+F(k+1)\phi(k)\varepsilon^0(k+1), \forall k\in N$$
$$F(k+1)=F(k)-\frac{F(k)\phi(k)\phi^T(k)F(k)}{1+\phi^T(k)F(k)\phi(k)}, \forall k\in N \quad (5)$$
$$\varepsilon^0(k+1)=y(k+1)-\hat{\theta}^T(k)\phi(t), \forall k\in N,$$

with the following initial conditions:

$$F(0)=\frac{1}{\delta}I=(GI)I, 0<\delta<1 \quad (6)$$

The estimated $\hat{\theta}(k)$ represents the parameters of the polynomial plant model and $\phi^T(k)$ represents the measures vector.

This approach allows the users to verify, and if is necessary, to calibrate algorithm's robustness. Next expression and Figure 8 present "disturbance-output" sensibility function.

$$S_{vy}(e^{j\omega}) \overset{def}{=} H_{vy}(e^{j\omega}) = \\ = \frac{A(e^{j\omega})S(e^{j\omega})}{A(e^{j\omega})S(e^{j\omega}) + B(e^{j\omega})R(e^{j\omega})}, \quad \forall \omega \in R \tag{7}$$

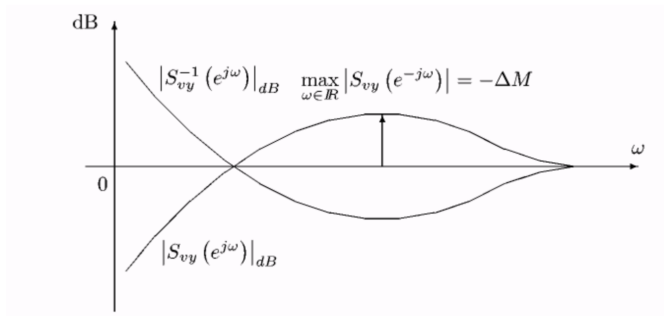

Fig. 8. Sensibility function graphic representation

At the same time, the negative maximum value of sensibility function represents the module margin.

$$\Delta M\big|_{dB} = -\max_{\omega \in R} \left|S_{vy}(e^{j\omega})\right|\Big|_{dB} \tag{8}$$

Base on this value [6], in a "input-output" representation, process nonlinearity can be bounded inside of "conic" sector, presented in Figure 9, where a_1 and a_2 are calculated using next expression:

$$\frac{1}{1-\Delta M} \geq a_1 \geq a_2 \geq \frac{1}{1+\Delta M} \tag{9}$$

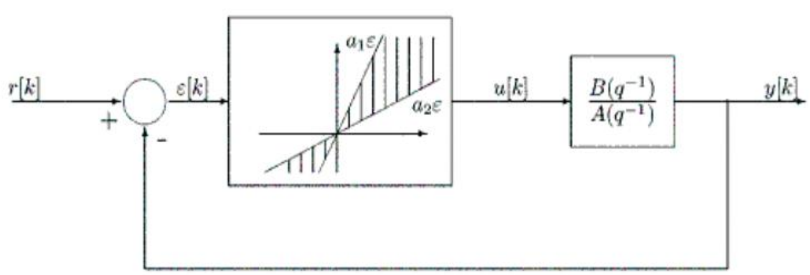

Fig. 9. Robust control design procedure

Finally, if it is imposed that all nonlinear characteristics should be (graphically) bounded by the two gains, or if the gain limit should be greater or equal to process static characteristic maximal distance ΔG≥*md*, then a controller that has sufficient robustness was designed.

3.4 Multi-model global architectures

Based on partitioning nonlinear characteristic like in Figure 4 and combining multi-model structure presented in Figure 1 with control structure presented in Figure 4 determine global architecture of multi-model control system presented in Figure 10.

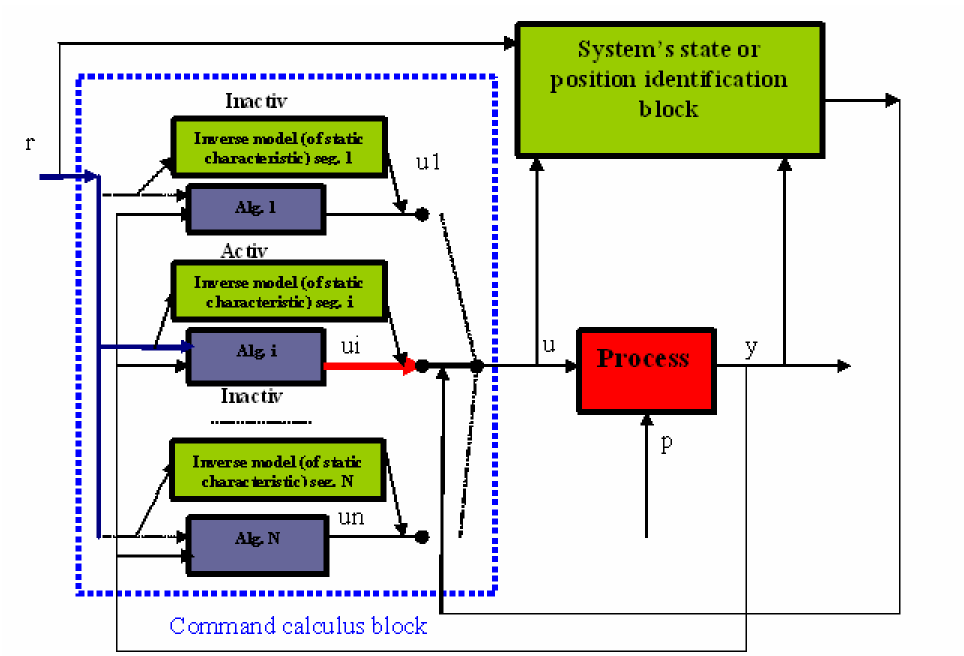

Fig. 10. Global architecture of multi-model control system

On Figure 10, the blocks and variables are as follows:

- Process – physical system to be controlled;
- Command calculus – unit that computes the process control law;
- Alg.i – i control algorithms (PID, RST);
- y – output of the process;
- u – output of the Command calculus block;
- u i. – output of the i control algorithm;
- r – system's set point or reference trajectory;
- p – disturbance of physical process.

4. Analysis of proposed structure

In this section we will present few advantages, disadvantages or limitation and some possible developing ways of presented structure.

4.1. Advantages of proposed structure

The main advantage consists in using a simplified and performing control structure based on classics procedure in designing of the control algorithm and determination of inverse command blocks comparative to classic multi-model control structure. There are used well know procedure for identification and control law design.

Because the global command contains a "constant" component generated by an inverse model command block, according to set point value, the system is very stabile.

Inverse model command generator can be replaced by a fuzzy logic bloc or neural network that can "contain" human experience about some nonlinear processes.

Because all the control laws are not very complex, real time software and hardware implementation don't need important resources.

4.2. Disadvantages or limitations of the structure

The main limitation is that this procedure can be applied just for the processes that support construction of static characteristics.

This structure is very difficult to use for the system that doesn't have a bijective characteristic and for systems with different functioning regimes.

Another limitation is that this structure can be used only for stabile processes. In situations where the process is "running", the direct (feed forward) command is very possible to not have enough flexibility to control it.

The increased number of experiments for determination of correct static characteristic can be other disadvantages of the structure.

4.3. Possible developing

For special situations, the direct command generators included in multi-model structure can be constructed as a single general block. This block compensate process nonlinearity and allow using simplified control laws in multiple controller structure.

These systems can be easily implemented on PLC structures particularly, and real time control systems, generally.

5. Experimental results

We have evaluated the achieved performances of the proposed control structure using an experimental installation presented in Figure 11, where the position of an object contained in the vertical tube must be controlled using an air flow generator.

Fig. 11. Experimental installation

The nonlinear process static characteristic is presented in Figure 12.

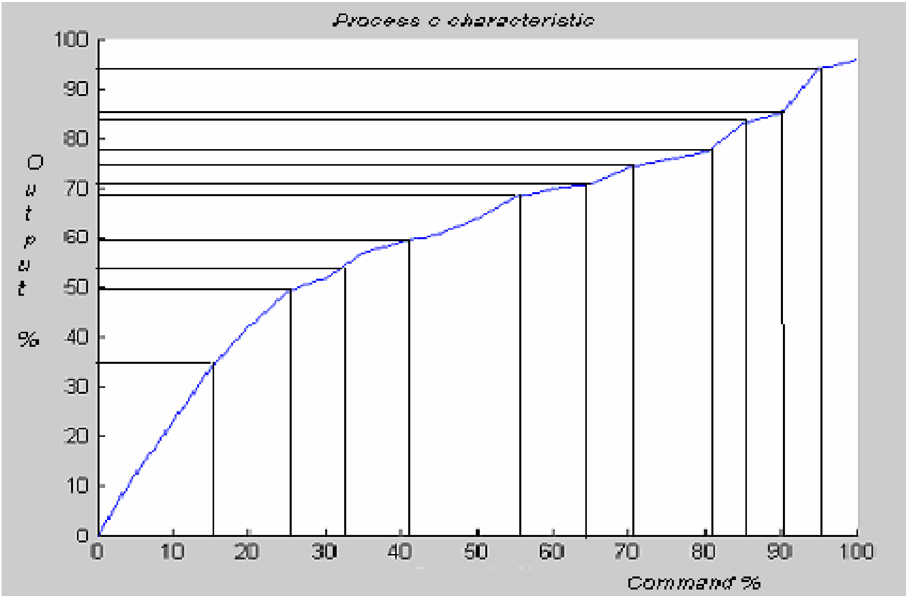

Fig. 12. Nonlinear process characteristic

For this, there are selected 12 zones (Figure 12) for a classic multi-model structure (Figure 1). The models and corresponding area (output %) are: M1: 0-35%, M2: 35-50%, M3: 50-54%, M4: 54-60%, M5: 60-69%, M6: 69-72%, M7: 72-75%, M8: 75-78%, M9: 78-84%, M10: 84-86%, M11: 86-95%, M12: 95-100%. All 12 models are first order complexity. For example, for M1 using Te=0.2 s sampling time and Least Square identification method from Adaptech/WinPIM the model is:

$$M_1 = \frac{0.487180}{1 - 0.79091 q^{-1}}$$

The corresponding controller (using pole placement): (for Tracking performances: second order dynamic system with w0=2.0, x=0.95, Disturbance rejection performances: second order dynamic system with w0=1.1, x=0.8, using WinReg):

$$R(q^{-1}) = 0.263281 - 0.179872\, q^{-1}$$
$$S(q^{-1}) = 1.000000 - 1.000000\ q^{-1}$$
$$T(q^{-1}) = 2.052629 - 3.412794 q^{-1} + 1.443573\ q^{-2}$$

The RST control algorithm can be written as follows:

$$u(k) = \frac{1}{s_0}[-\sum_{i=1}^{n_S} s_i u(k-i) - \\ -\sum_{i=0}^{n_R} r_i y(k-i) + \sum_{i=0}^{n_T} t_i y^*(k-i)]$$

where R, S, T polynomials are presented in relation (2) and n_S , n_R , n_T express the corresponding degrees and also the memory dimension for the software implementation of the algorithm. For example, if n_R=2, then it should be reserved three memory locations for the process's output: *y(k), y(k-1), y(k-2)*. Respectively, the same rule applies for *u(k)* and $y^*(k)$.

To calculate corresponding command for a single controller presented before, there are used 7 multiplies and 7 adding or subtraction operations.

Because the multi-models control structure must assure no bump commutations, all of 12 control algorithms work in parallel [7]. This condition gave the total number of operations 12 x 7 = 84 multiplies and 12 x 7 = 84 addling or subtraction operations.

For proposed control structure, with nonlinear blocks there are selected 3 "major" zones Z1: 0-50%, Z2: 50-80% and Z3: 80-100%, presented in Figure 13.

The models are:

$$M_1 = \frac{0.0964 - 0.19647q^{-1}}{1 - 1.06891q^{-1} + 0.22991q^{-2}}$$

$$M_2 = \frac{0.01297 + 0.05397q^{-1} + 0.03674q^{-2}}{1 - 0.76251q^{-1}}$$

$$M_3 = \frac{0.02187 + 0.05668q^{-1} + 0.06048q^{-2}}{1 - 0.93161q^{-1} + 0.02741q^{-2} + 0.09863q^{-3}}$$

In this case, we have computed three corresponding RST algorithms using a pole placement procedure from Adaptech/WinREG platform. The same nominal performances are imposed to all systems, through a second order system, defined by the dynamics $\omega_0 = 1.25$, $\xi = 1.2$ (tracking performances) and $\omega_0 = 2$, $\xi = 0.8$ (disturbance rejection performances) respectively, keeping the same sampling period as for identification.

$$R_1(q^{-1}) = 1.863259 - 2.027113q^{-1} + 0.520743q^{-2}$$
$$S_1(q^{-1}) = 1.000000 - 0.554998q^{-1} + 0.445002q^{-2}$$
$$T_1(q^{-1}) = 3.414484 - 4.931505q^{-1} + 1.873910q^{-2}$$

$$R_2(q^{-1}) = 2.309206 - 1.624937q^{-1}$$
$$S_2(q^{-1}) = 1.0 - 0.815278q^{-1} - 0.106427q^{-2} - 0.078295q^{-3}$$
$$T_2(q^{-1}) = 9.645062 - 14.928993q^{-1} + 5.968200q^{-2}$$

$$R_3(q^{-1}) = 1.72482 - 1.611292q^{-1} - 0.03784q^{-2} + 0.292903q^{-3}$$
$$S_3(q^{-1}) = 1.0 - 0.725187q^{-1} - 0.095205q^{-2} - 0.179608q^{-3}$$
$$T_3(q^{-1}) = 7.192692 - 11.645508q^{-1} + 4.821405q^{-2}$$

To calculate corresponding command for a C1 controller there are used 9 multiplies and 9 adding or subtraction operations, for C2 9 multiplies and 9 adding or subtraction operations and for C3 11 multiplies and 11 adding or subtraction operations, total number 29 multiplies and 29 adding or subtractions.

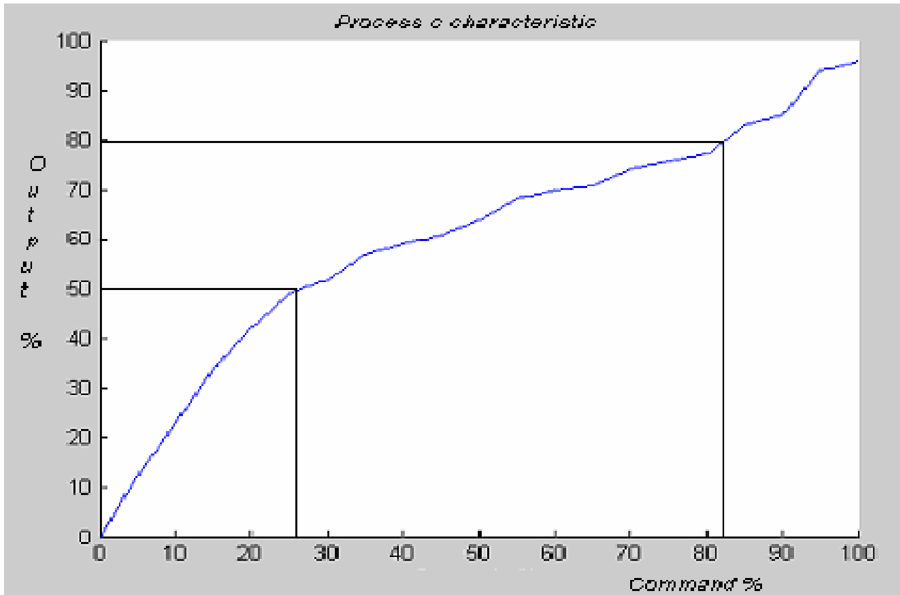

Fig. 13. Selection of the three zones of nonlinear characteristic

For the proposed control structure, in addition to command calculus operation, here is the calculus of direct command. This depends on software implementation. For PLC, particular and real time process computer, in general, where (C) code programming can be used, in a solution or other similar implementation:

```
// segment determination
segment = (int)(floor(rdk/10));
// segment gain and difference determination
panta = (tab_cp[segment+1] - tab_cp[segment]) * 0.1;
// linear value calculus
val_com_tr = uk + 1.00 * (panta * (rdk - segment*10.0) + tab_cp[segment]);
```

one needs 10 multiplying and 4 adding or subtraction operations (the time and memory addressing effort operation is considered equal to a multiplying operation). Total operations number for the proposed structure is 59 multiplying and 41 adding or subtraction operations.

It is obvious that proposed structure has a less number of multiplies comparative to classic multi-model solutions and a comparative value for the number of adding and subtractions. This means that the system with nonlinear compensator is faster or need a more simplified hardware and software architecture.

6. Conclusions

In this paper there is proposed a multi-model control structure which contains, for each model/controller, a nonlinearity compensator. This solution allows a reduced number of models and a reduced complexity for global structure. The main idea consists on determination of static characteristic for each model.

For this part of multi-model structure there are presented the design methods. These are based on experimental tests and classics identification and close loop pole placement.

There are made some analysis about advantages and disadvantages of proposed structure.

The experimental results effectuated on a experimental installation represent a case where proposed structure is a more fast solution comparative to classic multi-model structure. This structure can be easily implemented on PLC and real time process computer.

Acknowledgement

This work was supported by IDEI Research Program of Romanian Research, Development and Integration National Plan II, Grant no. 1044/2007.

R E F E R E N C E S

[1]. *K.S. Narendra, J. Balakrishnan*, Adaptive Control using multiple models, IEEE Transactions on Automatic Control, **vol. 42**, no. 2, pp. 171–187, Feb., 1997.
[2]. *I.D. Landau, A. Karimi*, Recursive algorithm for identification in closed loop: a unified approach and evaluation, Automatica, **vol. 33**, no 8, pp.1499-1523, 1997.
[3]. *L. Ljung, T. Soderstroom*, Theory and Practice of Recursive Identification, MIT Press, Cambridge, Massashusetts, 1983.
[4]. *D. Ştefãnoiu, J. Culiţă, P. Stoica*, - *Fundamentele Modelarii şi Identificării Sistemelor*, Editura Printech, ISBN 973-718-368-1, 2005.
[5]. *G. Tao, P. Kokotovic*, Adaptive control of systems with actuator and sensor nonlinearities, Wiley, N.Y., 1996.
[6]. *I.D. Landau., R. Lozano, M. M'Saad*, Adaptive Control, Springer Verlag, London, ISBN 3-540-76187-X, 1997.
[7]. *C. Lupu, D. Popescu, B. Ciubotaru, C. Petrescu, G. Florea* - Switching Solution for Multiple Models Control Systems, MED06, paper WLA2-2, Ancona, Italy, 28-30 June, 2006.
[8]. *I. Dumitrache*, Ingineria Reglării Automate, Politehnica Press, Bucureşti, 2005.
[9]. *M. Dussud, S. Galiche, L. Foulloy*, Fuzzy supervision for continuous casting mold level control, IFAC, 2000.
[10]. *J.M. Flaus*, La Regulation Industrielle, Editure Herms, Paris, 1994.

[11]. *I.D. Landau, K. Boumaiza*, An output error recursive algorithm for unbiased identification in closed-loop, IFAC Word Congress, San Francisco, 1996.
[12]. *O. Pages, P. Mouille, B. Caron*, Multi-model control by applying a symbolic fuzzy switches, IFAC, 2000.
[13]. *G. Pajunen*, Adaptive control of wiener type nonlinear system, Automatica, no. 28, pp. 781-785, 1992.
[14]. *J. Richalet*, Practique de la Commande Predictiv, Editura Herms, Paris, 1993.
[15]. *R.R Schrama, P. M. Van den Hof*, Accurate Identification for Control, the necessary of an iterative scheme, IEEE Transactions on Automatic Control, 1992.
[16]. *Z. Zang, R. R. Bitmead, M Gevers*, Iterative weighted LS Identification and LQG control design, Automatica, 1995.
[17]. *P.M. Van den Hof*, Closed loop issues in Systems Identification, IFAC, SYSID'97, Conference, Japan, 1997.