

New Management for the Control Three Tank System

MARIUS-CONSTANTIN POPESCU¹
LILIANA PERESCU-POPESCU²

GHEORGHE MANOLEA¹
NIKOS MASTORAKIS³

¹Faculty of Electromechanical and Environmental Engineering, University of Craiova
Decebal Bv, No.107, 200440, Craiova.

²“Elena Cuza” College of Craiova
ROMANIA.

³Technical University of Sofia
BULGARIA.

mrpopescu@em.ucv.ro,

ghmanolea@gmail.com

mastor@wse.org

Abstract: The development of fuzzy regulators designed for conduction of neminimum faze systems need a great volum of work conected to: defining the basic rules, chosing the values of free parameters of fuzzy regulators and checking the stability of the automated guiding systems with fuzzy regulators. The objectives of this paper consists in the application of FuzzyLib library regulators to develop quasi-fuzzy PI Mamdani-type equipment for laboratory management system with three tanks and the modification of software and interface to enhance the existing flexibility in the sense of providing the facility to develop more regulators.

Key-Words: Nonlinear mathematical model, Fuzzy regulators.

1 Introduction

Three tank system (3TS) is a model player for teaching that has been very used (and beloved) for study management solutions automatically. That the system is strongly nonlinear, with different possibilities of disturbance made this system to be present as a model for study in all laboratories of the Faculties of Automation reputation in Europe (Duisburg, Budapest, Vienna, Lund, etc.). To refine this system it was tried to develop a fuzzy regulator development to it. Fuzzy regulators belong to the class of general non-linear regulators [9]. So their input-output is defined by unambiguous feature static entry. Unfortunately, now there is no general method available for developing fuzzy crowds convenient and no adjustable fuzzy rules for solving a specific problem/application. Details of the modules of fuzzy regulators and fuzzy regulators development in of different structures are presented in [16]. Study of the various classes of fuzzy processes of interest from the regulator and from the system as a whole by analysing the effects of changes in process parameters led to the achievable performance of automatic systems operating on the basis of the proposed solutions [3], [4].

2 Presentation Three Tank System

2.1 Description of Three Tank System

The system is composed of three modules construction (parts), Fig.1 system with three tanks (process driven), including the execution (pumps) and measurement (sensors), the electronic element of performance actuator and plate DAC98 management.

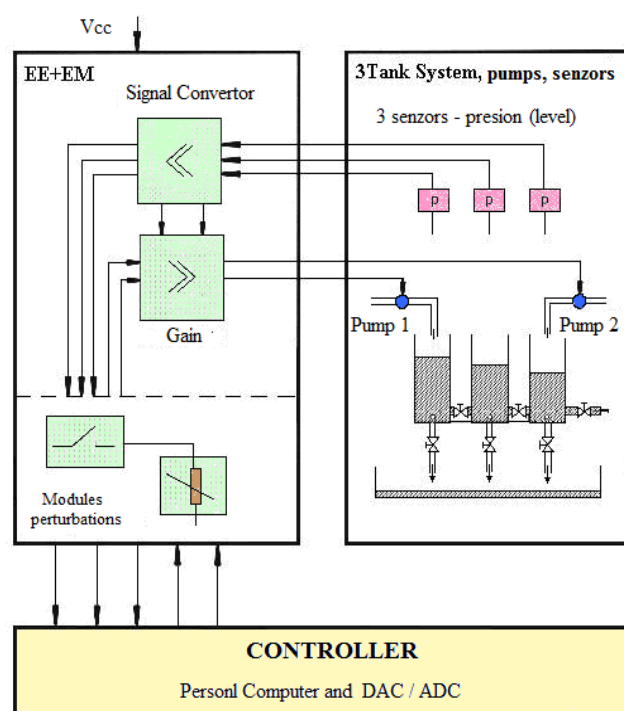


Fig. 1: Three tank system, inner scheme.

Leadership is provided by the management DTS200, which is supplied with the source design, allows for new software license. The maximum that can be attained is 60 cm (3 cm), after which - by the entry into service protection - pumps are disconnected automatically. Pumps are involved in DC-motor. For simulating disturbances in the system simulation, the interconnection piping and leakage rated valves are equipped with rosettes. At the same time each tank has a direct exit to the pool to collect the liquid. Measurements of the liquid levels are made for each tank by piezoresistive pressure transducers (Fig.2).

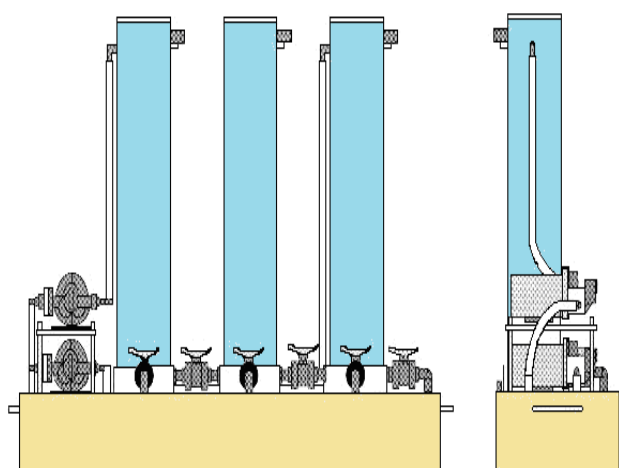


Fig. 2: Three tank system (the process).

The actuator includes the implementation and the measurements and is intended to convert the liquid (from the three tanks) in tension (to the numeric computer), but also to amplify the signal voltage from the computer in signal voltage to pumps. Part of the electronic element of execution is equipped with two internal plates (SERVO), one for each pump. For each of the pumps, there is the possibility of crossing the "Manual" mode "Automatic" and vice versa, respectively, with potentiometer P1 and P2 exist and can "control" the flow of the two pumps. The SENSOR allows measurements (blood) flow of two pumps (entry two elements of execution) and measuring the corresponding voltage level of the liquid tanks (measuring output elements). The SIGNAL ERROR is the additional "kit" of (option Opt.200-05). It is dedicated to the control system of the sensitivity errors measurement. With a potentiometer one can adjust (in percentages) the sensitivity for each tank separately and for each pump [2], [10], [11]. To avoid leakage of liquid outside the tank, where in the mode "Manual" is reached the limit of a reservoir, the electronic element of performance actuator was fitted with automatic stopping the pumping of liquid if the limit is reached (protection).

DAC98 management board is the interface with the regulatory automatic structure. This plate with the different "kit" sites of additional programs (in case the system is equipped with Opt.200-02, representing the program itself DTS200, Opt.200-03, representing the sources of leadership itself DTS200 and Opt.200-05, these being the disturbance power modules) will have various case studies on the analysis 3TS [2]. The plate is connected to the electronic element of performance (current) through a cable exit of 50 pins.

2.2 The DTS200 Management System 3TS

The management program is a 16-bit software with the default 50ms sampling, but adjustable between 1 and 54 ms. The interface was written in Borland Delphi 1.0 and the adjustment in Borland C++4.52. Both programs were provided with equipment (Opt.200-03). The package contains programs and files in May:

- DAC698.DRV - for access to the PC card;
- PLOT16.DLL - for graphical output;
- DTS200.HLP - help file;
- BC450RTL.DLL - library "run-time";
- TIMER16.DLL - call library that supports the functions of cyclic SERVICE.DLL.

Description of management program submitted under this subparagraph relates to the existing interface supplied by the firm. Interface created in the ninth paper will be presented in paragraph 4. Presentation is important because the new software and new interface is a generalization of those presented here. Interface features in this subparagraph shall be transferred to the new and expanded.

3 Library Fuzzylib for Regulators to Implement Fuzzy

Library `FuzzyLib` is built version `FuzzyLib FL801` [1] as a strictly hierarchical structure. As a programming language oriented on objects can achieve this, the library has been programmed using programming language C++ [14]. In C++ classes form the basis of object oriented programming. Typical use for classes is that they are used to define the nature of an object. In fact, C++ class is the basic unit of encapsulation [15].

Fuzzy crowd is the lowest level of the hierarchy mentioned. To this end, for the fuzzy crowd a separate class named `FuzzySet` was defined. Since almost all processing of the structure of fuzzy regulator using fuzzy variety, design class was conducted in conjunction with optimization and run

times, while providing a universe of discourse is as high as possible. Since these issues completed each other, it was necessary to find a compromise between time and operating flexibility. A "double" representation was chosen because it is supported directly by the arithmetic co-processors. Instead, to run the library it is recommended to use a computer that has good performance even in the representation of numbers. In the absence of an arithmetic co-processors, fuzzy library can be applied only in the simple calculations or offline. Representing numbers' double has the advantage in that it provides a field of virtually unlimited basis for fuzzy crowds.

Class `FuzzySet` [1] is represented by polygonal lines [16]. This line is cached in the form of pairs of numbers with values x/y , corresponding to the amount of farms/extent of the firm being related. An object of class `FuzzySet` can theoretically reach 32,768 of these values. The next level in the hierarchy of the library is represented by a fuzzy linguistic variable. This variable is included in class `FuzzyVar` [2]. A linguistic variable combines a lot of the same fuzzy universe of discourse and process elements of every type in the `FuzzySet`. The values of x/y of objects of type `FuzzySet` class `FuzzyVar` will be automatically extended so as crowds fuzzy all contain the same number of values x/y (about scaling the crowds). X coordinates of the x/y are identical. Regarding the running time, it is recommended to use as few values x/y possible (3-5 are enough) and to use the same x coordinates in the x/y of crowds that are grouped in a single linguistic variable. Fuzzy rules for adjustment are built in the form of types of objects of class `FuzzyRule` [1]. Class `FuzzyRule` combines linguistic variables of input and output with the terms of respecting linguistic syntax rules regulating fuzzy. For this purpose, to each rule must be generated each an object separately. `FuzzyRule` class includes functions to interpret the rules regulating fuzzy.

4 Implementation of New Management system 3TS

4.1 Solution Based on Using a Fuzzy Regulator of Non-PI

The integration of the RG-F itself may be introduced in two ways: on exit RG-F, resulting in so-called RG-F-PI with integration in an exit standard for entry RG-F, resulting in so-called RG-F-PI with integration in an entry standard. A disadvantage of the RG-F-PI-standard IE is the relatively high value increment command module, $|*u_k|$, where e_k Δe_k and

have the same sign, similar disadvantage RG-F-PI-II standard is the high command module, $|u_k|$, where e_k e_k and have the same sign. Disadvantages mentioned are poignant when, due to the positive zero in the transfer function of the running processes. Fuzzy regulator of non-PI-based rules as amended and distribution of membership functions for linguistic terms of the order is made for managing systems with non minimum value and is applicable for the version with integrated regulator and output for integration with the entry regulator [12], [13].

Regarding fuzzy quasi-regulator of the PI output /command, the standard variant (RG-F-quasi-standard IP-IE), it is characterized by dynamics that is inserted through (see Fig.3 where omitted to simplify the presentation scalar blocks).

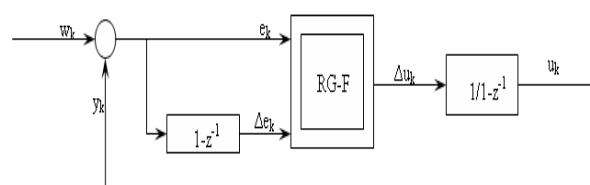


Fig. 3: The structure of an RG-F-quasi-PI-IE.

4.1.1 Changing the Management System 3TS, DTS200 for PI Variant with Regulators PI and other Regulators Fuzzy

Program management is a 16-bit software. The interface was written in Borland Delphi 1.0 and the adjustment in Borland C++4.52. Both programs were modified astfe to be used by users in their experiments. The package contains programs and files in May: DAC698.DRV (for access to the PC card), PLOT16.DLL (for graphical output), BC450RTL.DLL (library,run-time), TIMER16.DLL and SERVICE.DLL (part regulation and control). Part of the program files have been modified to achieve the adjustment and control. Of files added in addition to routine adjustment fuzzy and libraries to create new command SERVICE.DLL include `fuzzyCTL.cpp`, `ruleBase.fuz` and `fuzzy.lib`. In the program a modification was made to the graphical interface so that a better communication between the user interface can be achieved. Thus, the interface implemented by the producing company were introduced into the menu "Parameter", "Run" and "View" a series of submenus devoted most visible achievements of a fuzzy controller for the "Three Tank System". Run menu (Fig.5) includes adjustments to the system configuration. Thus were introduced submenus Fuzzy Controller and Reset Fuzzy Controller. Submenu Reset Fuzzy Controller installs a structure regulating the

fuzzy regulator. When another previously regulator is active, the generators of such default (for the liquid levels in tank 1 and 2) are reset.

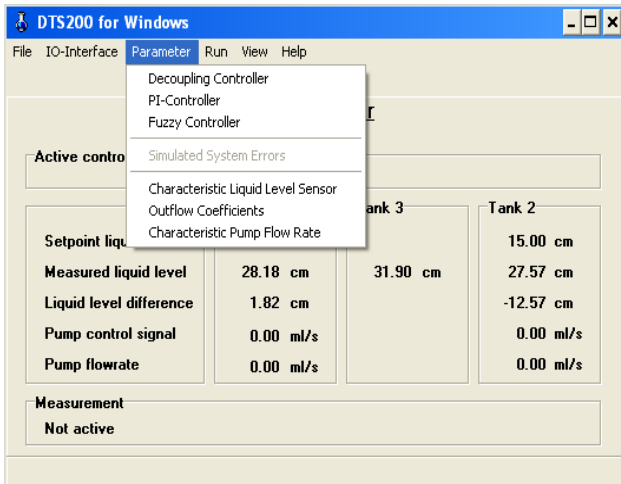


Fig. 4: Menu Parameter with sub-menu Fuzzy Controller.

Submenu called `Fuzzy Controller`, the two submenus `PI Reset Controller` and `PI Controller` to disable all set to eliminate any accidental activation of the two submenus, for reasons outlined above.

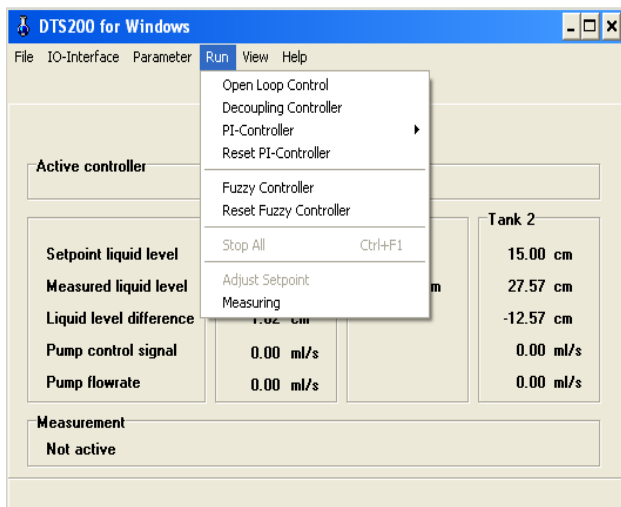


Fig. 5: Menu Run.

Submenu `Reset Fuzzy Controller` variables regulator resets the time of analysis. In the interface, so in Delphi 1.0, was a 3D graphics for fuzzy regulators corresponding to the two reservoirs. Thus in the View menu was added to a submenu `Fuzzy 3D`, which includes two or among options. Submenu `Fuzzy3D` was introduced in order to view the static characteristics of fuzzy regulators. By selecting one of the two options (`Controller fuzzy 1` or `Controller fuzzy 2`) will appear on-screen 3D graphics illustrated in Fig.6 [18]. Regarding the

programming in C++ to create file `fuzzyCTL.cpp` to use as interfacing between the rules `ruleBase.fuz` and `service.cpp` where the flow is calculated from the two pumps. Flows of the two pumps are then converted into the current so that according to a desired flow rate, it is more or less pumps are to send a current higher or lower.

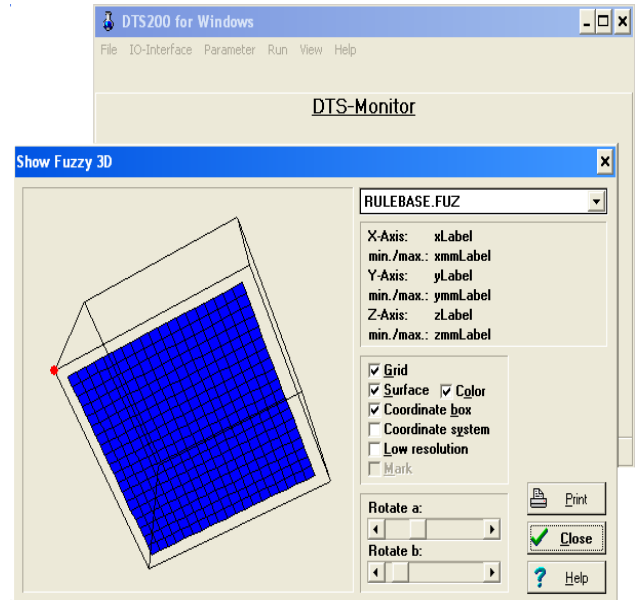


Fig. 6: Exemple static characteristic from fuzzy regulators.

`FuzzyCTL.h` file contains attributes and methods `CfuzzyController` class, the last will be implemented in the file `fuzzyCTL.cpp`. The data that are declared private are: `time`, `start` (the first step, when the error from the previous step initializes it with 0), `errors` (determining the number of errors occurred), `m_errorOldValue` (error value from the previous step), `m_fuzzy` (variable Fuzzy type) and `result` (the output). The methods are all public:

- `load` - load file descriptive fuzzy. Under this method is called `Fuzzy` class methods to implement fuzzy regulator: the class constructor method `generate()`. All errors occurred in the operation of obtaining a fuzzy object type (which will be in soft fuzzy regulator actually) are written into a file in the current directory which has the same name as the descriptive fuzzy (which is passed as a parameter function) but extension. `out`.

- `compute Parameters` - this method calculates the output u_k . RG uses a quasi-F-PI-IE. Derivation of numerical error adjustment in the form e_k increment error adjustment:

$$\Delta e_k = e_k - e_{k-1}, \tag{1}$$

is obtained as follows:

$$m_errorVariationValue = (m_errorCurrentValue$$

`-m_errorOldValue)/SAMPLETIME,`
 where `m_errorCurrentValue` value is the current error and send as parameter to this function, `m_errorOldValue` mean error value from the previous step and is `SAMPLETIME` sampling period. Incrementing number of increment Δu_k order to obtain such

`fuzzyOutputs *SAMPLETIME;`

where is the exit from `fuzzyOutputs` fuzzy controller [19].

- `resetController` –regulator;

- `GetName` returns the name-fuzzy descriptive

`char*CFuzzyController::getname(void)`

4.2 The Solution Based on Using Two Regulators PI

The main block diagram of the system with three tanks is shown in Fig.7. The basic relative reasons of the system were presented through the following guidelines: *e*, evacuation phenomenon of the liquid on the inside (through its on valves) and *g*, depletion phenomenon of the liquid on the outside (through the nominal valve). Tanks T_1 and T_2 are fed from pumps p_1 and p_2 with debits q_{p1} and q_{p2} mean while tank T_3 is being used as a tampon tank. The levels in the tanks are mesured with distributors TN1 (for T_1), TN2 (for T_2) and TN3 (for T_3). The levels from the tanks depend and the debits wich flow through the six faucets $\alpha_{e1}, \alpha_{e2}, \alpha_{e3}, \alpha_{s1}, \alpha_{s2}, \alpha_{s2}$:

- $\alpha_{e1}, \alpha_{e2}, \alpha_{e3}, \alpha_{g2}$ – of depletion (means disturbance)

- α_{s1}, α_{s2} – ensures trade of fluid between tanks $T_1 - T_3$ respectively $T_3 - T_2$.

The six faucets are identical, having section *S*.

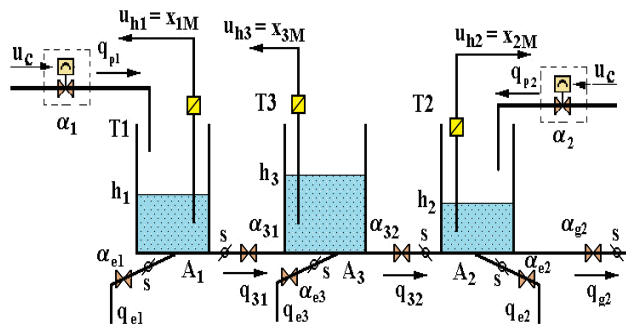


Fig. 7: Main block diagrame 3TS.

The purpose of regulation is constant maintaining of liquid levels in tanks T_1 and T_2 , h_1 and h_2 , through the debits of filling q_{p1} and q_{p2} and through the tampon tank T_3 . Disturbant effects are [6], [7], [8]:

- the level in tampon tank T_3 ;

- leakage of fluid through faucets $\alpha_{e1}, \alpha_{e2}, \alpha_{e3}, \alpha_{g2}$ marked through debits $q_{e1}, q_{e2}, q_{e3}, q_{20}$.

Trade debits q_{13} and q_{32} are being considered oriented: $q_{13} > 0$, if $h_1 > h_3$ and $q_{13} < 0$, if $h_1 < h_3$ respectively $q_{32} > 0$, if $h_3 > h_2$ and $q_{32} < 0$, if $h_3 < h_2$.

4.2.1 The Nonlinear Mathematical Model

Accepting the hypothesis of „uniform environment” and of a report between section of a faucets and section of a tank very small:

$$\frac{S}{A} \ll 1, \quad (2)$$

The flow speed of the liquid will be given by the formula [6]:

$$v \approx \sqrt{2g\Delta h}, \quad (3)$$

In wich Δh represents the difference between the interconnected tanks. For the two interconnected tanks ($T_1 - T_3$ respectively $T_3 - T_2$) the ecuation of Bernoulli is customized by this form :

$$v_{13} = \text{sgn}(h_1 - h_3) \sqrt{2g|h_1 - h_3|}, \quad (4)$$

with $v_{13} > 0$ for $h_1 > h_3$ (leakage $T_1 \rightarrow T_3$) or $v_{13} < 0$ for $h_1 < h_3$ (leakage $T_3 \rightarrow T_1$), respectively:

$$v_{32} = \text{sgn}(h_3 - h_2) \sqrt{2g|h_3 - h_2|}, \quad (5)$$

with $v_{32} > 0$ for $h_3 > h_2$ (leakage $T_3 \rightarrow T_2$) or $v_{32} < 0$ for $h_3 < h_2$ (leakage $T_2 \rightarrow T_3$) and where function *sgn* is wellknown mathematical function of signum:

$$\text{sgn}: \Re \rightarrow \{-1, 0, 1\}, \quad \text{sgn}(x) = \begin{cases} -1, & x < 0 \\ 0, & x = 0 \\ 1, & x > 0 \end{cases} \quad (6)$$

Concerning the volumetric balance of the liquid in each of these tanks is given by the formula:

$$\dot{h}_i A_i = \sum [q_{iin} - q_{iout}] = q_{\Sigma i}, \quad i = \{1, 2, 3\}. \quad (7)$$

In particular for each tank the formula (7) can be written ($A_1 = A_2 = A_3 = A$):

$$\mathbf{T}_1: \dot{h}_1 = \frac{1}{A} [q_{p1} - q_{13} - q_{e1}], \quad (8)$$

$$\mathbf{T}_2: \dot{h}_2 = \frac{1}{A} [q_{p2} + q_{32} - q_{e2} - q_{20}], \quad (9)$$

$$\mathbf{T}_3: \dot{h}_3 = \frac{1}{A} [q_{13} - q_{32} - q_{e3}] , \quad (10)$$

In wich the debits have the expressions:

$$q_{13} = \mu_{S1} \cdot S \cdot \operatorname{sgn}(h_1 - h_3) \sqrt{2g|h_1 - h_3|} \quad (11)$$

$$q_{32} = \mu_{S2} \cdot S \cdot \operatorname{sgn}(h_3 - h_2) \sqrt{2g|h_3 - h_2|} , \quad (12)$$

$$q_{20} = \mu_{g2} \cdot S \cdot \sqrt{2gh_2} , \quad (13)$$

$$q_{ei} = \mu_{ei} \cdot S \cdot \sqrt{2gh_i} , \quad i = \{1,2,3\} \quad (14)$$

$$q_{pi} = c_i u_{ci} , \quad i = \{1,2\} \quad (15)$$

The leakage q_{ei} can be treated as „mallfunctions” in the systems. The parameters $\mu_{S1}, \mu_{S2}, \mu_{g2}, \mu_{ei}$ represent the multipliers of proportionality, with $\mu \in [0,1]$, where $\mu = 0, \mu = 1$ - completley shut off faucet, respectively, completley open. In formula (15) u_1 and u_2 represent the commands given to the two pumps. For simplicity will consider products $\mu_i \cdot S = u_i$ ($i = \{S_1, S_2, g_2, e_1, e_2, e_3\}$). Reuniting formulas (8-10) and (11-15) will obtain the alinear mathematical model of the sistem 3TS:

$$\dot{h}_1 = \frac{1}{A} u_{s1} \cdot \operatorname{sgn}(h_1 - h_3) \cdot \sqrt{2g|h_1 - h_3|} - \frac{1}{A} u_{e1} \sqrt{2gh_1} + \frac{1}{A} c_1 u_{c1} , \quad (16)$$

$$\dot{h}_2 = \frac{1}{A} u_{s1} \cdot \operatorname{sgn}(h_3 - h_2) \cdot \sqrt{2g|h_3 - h_2|} - \frac{1}{A} u_{e2} \sqrt{2gh_2} + \frac{1}{A} u_{g2} \sqrt{2gh_2} + \frac{1}{A} c_2 u_{c2} , \quad (17)$$

$$\dot{h}_3 = \frac{1}{A} u_{s1} \cdot \operatorname{sgn}(h_1 - h_3) \cdot \sqrt{2g|h_1 - h_3|} - \frac{1}{A} u_{s2} \operatorname{sgn}(h_3 - h_2) \sqrt{2g|h_3 - h_2|} - \frac{1}{A} u_{e3} \sqrt{2gh_3} . \quad (18)$$

For $S/A \ll 1$ report, the model is deductible with good results in a arbitrary static functionality point A_0 of coordinates $\{h_{10}, h_{20}, h_{30}, u_{c10}, u_{c20}, \mu_{S10}, \mu_{S20}, \mu_{e10}, \mu_{e20}, \mu_{e30}, \mu_{g20}\}$.

4.2.2 Linearisation of the Mathematical Model

As it is obvious, the mathematical model presented in formulas (16-18) is a mathematical alinear model. For the linearisation of the model we use the tangent method. Linearitisation is done equation with equation. So for the first equation (16):

$$\dot{h}_1 = \frac{1}{A} u_{s1} \cdot \operatorname{sgn}(h_1 - h_3) \cdot \sqrt{2g|h_1 - h_3|} - \frac{1}{A} u_{e1} \sqrt{2gh_1} + \frac{1}{A} c_1 u_{c1} \quad (19)$$

It is noted $T_{11} = \frac{\sqrt{2g}}{A} \mu_1 \cdot S \cdot \operatorname{sgn}(h_1 - h_3) \sqrt{|h_1 - h_3|}$,

where $\mu_1 S = u_{S1}$. As we can see there is a discution regarding the sign of function T_{11} depending on the signum function value:

a) It is considered $h_1 > h_3$ for wich $\operatorname{sgn}(h_1 - h_3) = 1$ and we obtain [8]:

$$\Delta T'_{11} = \frac{\sqrt{2g}}{A} \sqrt{h_{10} - h_{30}} \cdot \Delta u_{S1} + \frac{\sqrt{2g}}{A} u_{S10} \frac{1}{2\sqrt{h_{10} - h_{30}}} \Delta h_1 - \frac{\sqrt{2g}}{A} u_{S10} \frac{1}{2\sqrt{h_{10} - h_{30}}} \Delta h_3 \quad (20)$$

b) If $h_1 < h_3 \Rightarrow \operatorname{sgn}(h_1 - h_3) = -1$ we will obtain

$$\Delta T'_{11} = -\frac{\sqrt{2g}}{A} \sqrt{h_{30} - h_{10}} \Delta u_{S1} - \frac{\sqrt{2g}}{A} u_{S10} \frac{1}{2\sqrt{h_{30} - h_{10}}} \Delta h_3 + \frac{\sqrt{2g}}{A} u_{S10} \frac{1}{2\sqrt{h_{30} - h_{10}}} \Delta h_1 \quad (21)$$

Besides the discution regarding the sign of equation (20) and (21) present the necessity of a discution looking the effective difference $\Delta(h_1 - h_3)$:

A) $h_{10} > h_{30}$

$$\left. \begin{matrix} \Delta h_1 = h_1 - h_{10} < 0 \\ \Delta h_3 = h_3 - h_{30} < 0 \end{matrix} \right\} \Rightarrow |\Delta h_1| > |\Delta h_3| \Rightarrow \Delta(h_1 - h_3) > 0$$

$$h_1 > h_3$$

B) $h_{10} > h_{30} \Rightarrow \left. \begin{matrix} \Delta h_1 < \Delta h_3 \\ h_1 > h_3 \end{matrix} \right\} \Rightarrow \Delta(h_1 - h_3) > 0$

C) $h_{10} > h_{30} \Rightarrow \left. \begin{matrix} \Delta h_1 > \Delta h_3 \\ h_1 < h_3 \end{matrix} \right\} \Delta(h_1 - h_3) < 0$

We linearise then the second factor from equation

$$(19) T_{12} = \frac{\sqrt{2g}}{A} u_{e1} \sqrt{h_1} , \quad \text{where } u_{e1} = \mu_{e1} \cdot S \quad \text{from}$$

where:

$$\Delta T_{12} = \frac{\sqrt{2g}}{A} u_{e1} \sqrt{h_{10}} + \frac{\sqrt{2g}}{A} u_{e10} \frac{\sqrt{1}}{2\sqrt{h_{10}}} \Delta h_1 . \quad (22)$$

Analog for the third term $T_{13} = \frac{1}{A} c_1 u_{c1}$, or :

$$\Delta T_{13} = \frac{c_1}{A} \Delta u_{c1} \quad (23)$$

Reuniting the linearised terms (20), (22) and (23) in the first equation of state (19) we obtain the linearisation of the first equation of state:

$$\begin{aligned} \Delta \dot{h}_1 = & -\Delta T_{11} - \Delta T_{12} + \Delta T_{13}, \quad (24) \\ \Delta \dot{h}_1 = & -\frac{\sqrt{2g}}{A} \operatorname{sgn}(h_{10} - h_{30}) \cdot \sqrt{|h_{10} - h_{30}|} \cdot \Delta u_{s1} - \\ & \frac{\sqrt{2g}}{A} u_{s10} \operatorname{sgn}(h_{10} - h_{30}) \frac{1}{2\sqrt{|h_{10} - h_{30}|}} \Delta h_1 + \\ & + \frac{\sqrt{2g}}{A} u_{s10} \operatorname{sgn}(h_{10} - h_{30}) \frac{1}{2\sqrt{|h_{10} - h_{30}|}} \Delta h_3 - \\ & - \frac{\sqrt{2g}}{A} \sqrt{h_{10}} \cdot \Delta u_{e1} - \frac{\sqrt{2g}}{A} u_{e10} \frac{1}{2\sqrt{h_{10}}} \Delta h_1 + \frac{c_1}{A} \Delta u_{c1} \quad (25) \end{aligned}$$

Then follows the linearisation of the second equation of state of the ailinear model, (17):

$$\begin{aligned} \dot{h}_2 = & \frac{1}{A} u_{s1} \cdot \operatorname{sgn}(h_3 - h_2) \cdot \sqrt{2g|h_3 - h_2|} - \\ & \frac{1}{A} u_{e2} \sqrt{2gh_2} + \frac{1}{A} u_{g2} \sqrt{2gh_2} + \frac{1}{A} c_2 u_{c2} \quad (26) \end{aligned}$$

In wich there are the following notations: $\mu_2 S = u_{s2}$, $\mu_{e2} S = u_{e2}$, $\mu_{g2} S = u_{g2}$. The liniarisation is done in this case on terms, and the discution concerning the sign of the terms are maintaned just like the first equation of state. This way we obtain:

$$\begin{aligned} \Delta T_{21} = & \frac{\sqrt{2g}}{A} u_{s20} \operatorname{sgn}(h_{30} - h_{20}) \cdot \frac{1}{2\sqrt{|h_{30} - h_{20}|}} \Delta h_3 - \\ & - \frac{\sqrt{2g}}{A} u_{s20} \operatorname{sgn}(h_{30} - h_{20}) \cdot \frac{1}{2\sqrt{|h_{30} - h_{20}|}} \Delta h_2 + \quad (27) \end{aligned}$$

$$+ \frac{\sqrt{2g}}{A} \operatorname{sgn}(h_{30} - h_{20}) \sqrt{|h_{30} - h_{20}|} \Delta u_{s1}$$

$$\Delta T_{22} = \frac{\sqrt{2g}}{A} u_{e20} \frac{1}{2\sqrt{h_{20}}} \Delta h_2 + \frac{\sqrt{2g}}{A} \sqrt{h_{20}} \Delta u_{e2}, \quad (28)$$

$$\Delta T_{23} = \frac{\sqrt{2g}}{A} u_{g20} \frac{1}{2\sqrt{h_{20}}} \Delta h_2 + \frac{\sqrt{2g}}{A} \sqrt{h_{20}} \Delta u_{g2} \quad (29)$$

$$\Delta T_{24} = \frac{c_2}{A} \Delta u_{c2} \quad (30)$$

Reuniting equations (27-30) after equation (26), we obtain the linearised form for the second equation of state:

$$\Delta \dot{h}_2 = \Delta T_{21} - \Delta T_{22} - \Delta T_{23} + \Delta T_{24}, \quad (31)$$

$$\begin{aligned} \Delta \dot{h}_2 = & \frac{\sqrt{2g}}{A} \operatorname{sgn}(h_{30} - h_{20}) \cdot \sqrt{|h_{30} - h_{20}|} \cdot \Delta u_{s2} + \\ & \frac{\sqrt{2g}}{A} u_{s20} \cdot \operatorname{sgn}(h_{30} - h_{20}) \frac{1}{2\sqrt{|h_{30} - h_{20}|}} \Delta h_3 - \\ & - \frac{\sqrt{2g}}{A} u_{s20} \operatorname{sgn}(h_{30} - h_{20}) \frac{1}{2\sqrt{|h_{30} - h_{20}|}} \Delta h_2 - \\ & \frac{\sqrt{2g}}{A} \sqrt{h_{20}} \cdot \Delta u_{e2} - \frac{\sqrt{2g}}{A} u_{e20} \frac{1}{2\sqrt{h_{20}}} \Delta h_2 - \\ & - \frac{\sqrt{2g}}{A} \sqrt{h_{20}} \cdot \Delta u_{g2} - \frac{\sqrt{2g}}{A} u_{g20} \frac{1}{2\sqrt{h_{20}}} \Delta h_2 + \frac{c_2}{A} \Delta u_{c2} \quad (32) \end{aligned}$$

Finally we make the linearization of the third equation of state on the same principles/discussions like the first two:

$$\begin{aligned} \dot{h}_3 = & \frac{1}{A} u_{s1} \cdot \operatorname{sgn}(h_1 - h_3) \cdot \sqrt{2g|h_1 - h_3|} - \\ & \frac{1}{A} u_{s2} \operatorname{sgn}(h_3 - h_2) \sqrt{2g|h_3 - h_2|} - \frac{1}{A} u_{e3} \sqrt{2gh_3} \quad (33) \end{aligned}$$

Linearising term by term and introducing notations: $\mu_1 S = u_{s1}$, $\mu_2 S = u_{s1}$, $\mu_{e3} S = u_{e3}$, we obtain:

$$\begin{aligned} \Delta T_{31} = & \frac{\sqrt{2g}}{A} u_{s10} \operatorname{sgn}(h_{10} - h_{30}) \frac{1}{2\sqrt{|h_{10} - h_{30}|}} \Delta h_1 - \\ & - \frac{\sqrt{2g}}{A} u_{s10} \operatorname{sgn}(h_{10} - h_{30}) \frac{1}{2\sqrt{|h_{10} - h_{30}|}} \Delta h_3 + \quad (34) \end{aligned}$$

$$+ \frac{\sqrt{2g}}{A} \operatorname{sgn}(h_{10} - h_{30}) \sqrt{|h_{10} - h_{30}|} \Delta u_{s1}$$

$$\begin{aligned} \Delta T_{32} = & \frac{\sqrt{2g}}{A} u_{s20} \operatorname{sgn}(h_{30} - h_{20}) \frac{1}{2\sqrt{|h_{30} - h_{20}|}} \Delta h_3 - \\ & - \frac{\sqrt{2g}}{A} u_{s20} \operatorname{sgn}(h_{30} - h_{20}) \frac{1}{2\sqrt{|h_{30} - h_{20}|}} \Delta h_2 +, \quad (35) \end{aligned}$$

$$+ \frac{\sqrt{2g}}{A} \operatorname{sgn}(h_{30} - h_{20}) \sqrt{|h_{30} - h_{20}|} \Delta u_{s2}$$

$$\Delta T_{33} = \frac{\sqrt{2g}}{A} u_{e30} \frac{1}{2\sqrt{h_{30}}} \Delta h_3 + \frac{\sqrt{2g}}{A} \sqrt{h_{30}} \Delta u_{e3}. \quad (36)$$

Reuniting equations (34-36) we obtain the linear form for the third equation of state:

$$\Delta \dot{h}_3 = \Delta T_{31} - \Delta T_{32} - \Delta T_{33}, \quad (37)$$

$$\begin{aligned} \Delta \dot{h}_3 = & \frac{\sqrt{2g}}{A} \operatorname{sgn}(h_{10} - h_{30}) \cdot \sqrt{|h_{10} - h_{30}|} \cdot \Delta u_{s1} + \\ & \frac{\sqrt{2g}}{A} u_{s10} \cdot \operatorname{sgn}(h_{10} - h_{30}) \frac{1}{2\sqrt{|h_{10} - h_{30}|}} \Delta h_1 - \\ & - \frac{\sqrt{2g}}{A} u_{s10} \operatorname{sgn}(h_{10} - h_{30}) \frac{1}{2\sqrt{|h_{10} - h_{30}|}} \Delta h_3 - \\ & \frac{\sqrt{2g}}{A} \operatorname{sgn}(h_{10} - h_{30}) \cdot \sqrt{|h_{10} - h_{30}|} \cdot \Delta u_{s2} - \\ & - \frac{\sqrt{2g}}{A} u_{s20} \operatorname{sgn}(h_{30} - h_{20}) \frac{1}{2\sqrt{|h_{30} - h_{20}|}} \Delta h_3 + \\ & \frac{\sqrt{2g}}{A} u_{s20} \operatorname{sgn}(h_{30} - h_{20}) \frac{1}{2\sqrt{|h_{30} - h_{20}|}} \Delta h_2 - \\ & - \frac{\sqrt{2g}}{A} \sqrt{h_{30}} \cdot \Delta u_{e3} - \frac{\sqrt{2g}}{A} u_{e30} \frac{1}{2\sqrt{h_{30}}} \Delta h_3 \quad (38) \end{aligned}$$

By this the mathematical linearised model is:

$$\begin{aligned} \Delta \dot{h}_1 = & -\frac{\sqrt{2g}}{A} \operatorname{sgn}(h_{10} - h_{30}) \cdot \sqrt{|h_{10} - h_{30}|} \cdot \Delta u_{s1} - \\ & \frac{\sqrt{2g}}{A} u_{s10} \operatorname{sgn}(h_{10} - h_{30}) \frac{1}{2\sqrt{|h_{10} - h_{30}|}} \Delta h_1 + \\ & + \frac{\sqrt{2g}}{A} u_{s10} \operatorname{sgn}(h_{10} - h_{30}) \frac{1}{2\sqrt{|h_{10} - h_{30}|}} \Delta h_3 - \\ & - \frac{\sqrt{2g}}{A} \sqrt{h_{10}} \cdot \Delta u_{e1} - \frac{\sqrt{2g}}{A} u_{e10} \frac{1}{2\sqrt{h_{10}}} \Delta h_1 + \frac{c_1}{A} \Delta u_{e1} \quad (39) \\ \Delta \dot{h}_2 = & \frac{\sqrt{2g}}{A} \operatorname{sgn}(h_{30} - h_{20}) \cdot \sqrt{|h_{30} - h_{20}|} \cdot \Delta u_{s2} + \\ & \frac{\sqrt{2g}}{A} u_{s20} \cdot \operatorname{sgn}(h_{30} - h_{20}) \frac{1}{2\sqrt{|h_{30} - h_{20}|}} \Delta h_3 - \\ & - \frac{\sqrt{2g}}{A} u_{s20} \operatorname{sgn}(h_{30} - h_{20}) \frac{1}{2\sqrt{|h_{30} - h_{20}|}} \Delta h_2 - \\ & \frac{\sqrt{2g}}{A} \sqrt{h_{20}} \cdot \Delta u_{e2} - \frac{\sqrt{2g}}{A} u_{e20} \frac{1}{2\sqrt{h_{20}}} \Delta h_2 - \end{aligned}$$

$$- \frac{\sqrt{2g}}{A} \sqrt{h_{20}} \cdot \Delta u_{g2} - \frac{\sqrt{2g}}{A} u_{g20} \frac{1}{2\sqrt{h_{20}}} \Delta h_2 + \frac{c_2}{A} \Delta u_{e2} \quad (40)$$

$$\begin{aligned} \Delta \dot{h}_3 = & \frac{\sqrt{2g}}{A} \operatorname{sgn}(h_{10} - h_{30}) \cdot \sqrt{|h_{10} - h_{30}|} \cdot \Delta u_{s1} + \\ & \frac{\sqrt{2g}}{A} u_{s10} \cdot \operatorname{sgn}(h_{10} - h_{30}) \frac{1}{2\sqrt{|h_{10} - h_{30}|}} \Delta h_1 - \\ & - \frac{\sqrt{2g}}{A} u_{s10} \operatorname{sgn}(h_{10} - h_{30}) \frac{1}{2\sqrt{|h_{10} - h_{30}|}} \Delta h_3 - \\ & \frac{\sqrt{2g}}{A} \operatorname{sgn}(h_{10} - h_{30}) \cdot \sqrt{|h_{10} - h_{30}|} \cdot \Delta u_{s2} - \\ & - \frac{\sqrt{2g}}{A} u_{s20} \operatorname{sgn}(h_{30} - h_{20}) \frac{1}{2\sqrt{|h_{30} - h_{20}|}} \Delta h_3 + \\ & \frac{\sqrt{2g}}{A} u_{s20} \operatorname{sgn}(h_{30} - h_{20}) \frac{1}{2\sqrt{|h_{30} - h_{20}|}} \Delta h_2 - \\ & - \frac{\sqrt{2g}}{A} \sqrt{h_{30}} \cdot \Delta u_{e3} - \frac{\sqrt{2g}}{A} u_{e30} \frac{1}{2\sqrt{h_{30}}} \Delta h_3 \quad (41) \end{aligned}$$

The mathematical model obtained complicates in the situation in which h_{i0} and h_{j0} are slightly equal and there is a passing through $h_i=h_j$. In this case the linearised model can lead to failure. Through linearization is decided that

$$\operatorname{sgn}(h_i - h_j) = \operatorname{sgn}(h_{i0} - h_{j0}), \quad i, j = \{1, 2, 3\};$$

The discussion concerning $\Delta(h_3 - h_2)$ needs to be considered like wise the discussion had at $\Delta(h_1 - h_3)$. The vector of disturbance - like it can be seen from the linearised model - is:

$$v = [\Delta u_{S1} \quad \Delta u_{S2} \quad \Delta u_{e1} \quad \Delta u_{e2} \quad \Delta u_{e3} \quad \Delta u_{g2}]^T.$$

4.2.3 Changing the Soft for the New Requirements

In the received software from the production firm the problem of the PI controller is treated for each tank individually, but the interface as well as the program transmit the same data. The two parameters from the adjustment equation implemented by the producer P_i and K_i , are the same for both regulators. In the modified version the user has for the first time the chance to transmit to the system two sets of different parameters, P_{i1} , P_{i2} , K_{i1} , K_{i2} . Also the interface allows separated functioning of the two pumps depending on the given parameters. It is allowed to start any pump, and then at a chosen interval of the

user starting of the second pump. Optionally we can start both pumps once by activating submenu "Run Both". We realised changes both in the interface aswell as in the C++ part, but following continuously the given code by the production firm. Thus to be able to introduce different parameters for the two regulators, appropriate to the two pumps, file `dlls.pas`, which keeps global data has been modified. Other modifications were made to the run menu which allows the simultaneous starting of the regulators according to Fig.8 [5], [17].

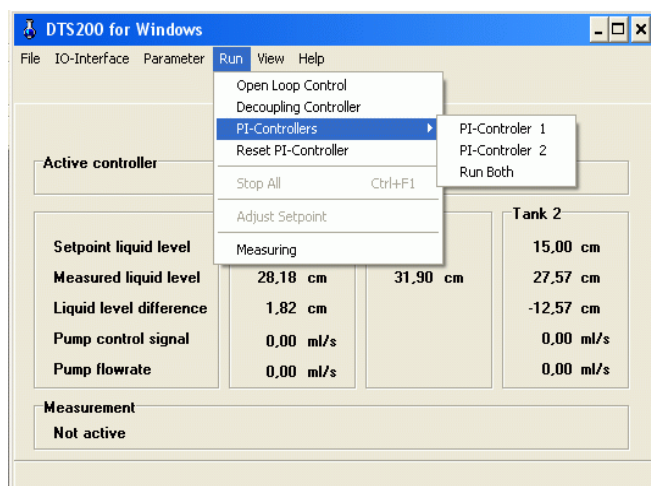


Fig. 8: Menu „Run,, with given modification.

As an observation the submenu `PI-Controllers` allows the starting of any regulator, but it won't allow but stopping of one single regulator or after starting one to try activating submenu "Run Both". Aswell, the frame "Active controller" will detect the presence of `PI-Controller 1` or `PI-Controller 2`, and will detect every time the two regulators will work at the same time even if these two will work together when activating submenu "Run Both" or when activating simultaneously.

4.2.4 Generating of New Programs

For a previous development stimulation of the two variants aswell as the whole system were kept the two variants program sources aswell as the sources given by the production firm, this way, making possible generating of new execution programs. The only restriction is to have Delphi programs (to create the interface) and Borland C++ (for the adjustment of DLL files). For the generation of the new programs it's necessary to copy the disks on the hard-disk, by following the same subdirector structures from the disks. This way after copying, we need to find directors: `DTSDESK` (contains file *Delphi Project DTS200W.DPR* with all files source Pascal for generating the interface), `CONTRDLL` (contains file *Borland Project DTSDLL.IDE*

together with all the sources files C++ to create the new file DLL of adjustment - `SOURCE.DLL`) and EXE (contains all additional files needed for the executable program - `DTS22.HLP`, `DTSW16.INI`, `DEFAULT.CAL`, `DAC*.DRV`, `TIMER16.DLL`, `PLOT16.DLL`, `BC450.DLL`). After creating a new package of programs (interface and/or adjustment - DLL), these need to be copied in subdirector EXE. In the attempt to create a new programs packet (specially of adjustment - DLL), will need to pay extra attention at library `fuzzy.lib` which might need to be reruned

5 Conclusions

If the existing interface allows implementation of two PI regulators with the same regulation for two tanks of 3TS new interface implementation allows two separate PI regulators to operate as separate systems for automatic monovariate level of one tank, as together as systems of multivariate automatic. In addition, the interface allows implementation of two separate regulators fuzzy quasi-PI. In this way a multidisciplinary paper was obtained with work that includes knowledge of hardware and software and setting. A fuzzy regulator per-say represents a alinear element, without dynamics. Dynamics is practically introduced on the outside of the fuzzy regulator per-say in two ways: dynamic refinement of the entrance sizes (prior fuzzyfication) or dynamic refinement of the exit (command)(after defuzzyfication). Minimum phase systems show some properties which are not convenient in leading like for example the characteristic phase-pulsation strongly drops with pulsation growth behaviour in time domain is deflecting from the usual behaviour of the minimum phase processes. As drawbacks of the fuzzy conduction we mention: it needs experience from the designer regarding selection of some free parameters of RG-F so it obtain the best behaviour of automated adjustment with RG-F, due to high level of calculus, implementation is more difficult compared with implementation of conventional regulators. Fuzzy regulators (RG-F) with dynamic (type Mandami) show the following advantages: they are easy to design; they don't need an exact mathematical modeling of the process; they ensure dynamic performance and stationary very good in the automated adjustment in which are framed, higher ensured performances of conventional regulators; as opposed to conventional regulators behaviour, they can ensure automated adjustment in which a better behaviour is frame for the parameter modifications

to the conducted process. Due to the number of parameters from the fuzzy regulators structures to develop the fuzzy regulators (RG-F) and giving these parameters is necessary the usage of adequate developments environments. The mathematical model of the conducted process from the frame "three tank systems" is multivariable of third level and strongly a linear which brings difficulties in stabilisation and adjustment in adjustment issues of the liquid level from the two exterior tanks (tanks 1 and 2). Fuzzy regulators after the exit (after adjustment error) or after state can satisfy in these conditions compared to other conduction. Although full of contradictions spectacular developments of the fuzzy conduction of the environment needs to be searched in more directions like: fuzzy conduction which ensures the adequate performances where price of some "special" performances becomes very high: fuzzy conduction can be abstract of mathematical modelling and of special mathematical development, which in some applications leaves open field to "engineering empiric"; fuzzy conduction can be implemented in more applications with relatively low prices, more convenient than using other conduction structures.

References

- [1] Amira, *FL801: FuzzyLib, Fuzzy Library*, Amira GmbH, Duisburg, 1997.
- [2] Amira, *DTS200: Laboratory Setup Three-Tank-System*, Amira GmbH, Duisburg, 1998.
- [3] Drighiciu M., Petrisor A., Popescu M.C., *A Petri Nets approach for hybrid systems modelling*, International Journal of Circuits, Systems and Signal Processing, Issue 2, Vol.3, pp.55-64, 2009.
- [4] FCM, *User's Manual*, TransferTech GmbH, Braunschweig, Germania, 1996.
- [5] FIDE, *User's Manual*, Apronix, SUA, 1996.
- [6] Matlab, *Fuzzy Logic Toolbox, User's Guide*, MathWorks Inc., Natick, MA, 2002.
- [7] Olaru O., Popescu M.C., Balas V. *A Study of Oscillation for Signal Stabilization of Nonlinear System*, Proceedings of the 10th WSEAS Int. Conf. on Automation & Information, pp.430-437, Prague, March 2009.
- [8] Petrișor A., Bizdoacă N.G., Drighiciu M., Popescu M.C., *Control Strategy of a 3-DOF Walking Robot*, The International Conference on „Computer as a Tool”, pp.2337-2342, Warsaw, Polonia, September 2007.
- [9] Popescu M.C., Olaru O., Mastorakis N., *Equilibrium Dynamic Systems Intelligence*, WSEAS Transactions on Information Science and Applications, Issue 5, Vol.6, pp.725-735, May 2009.
- [10] Popescu M.C., *Hybrid neural network for prediction of process parameters in injection moulding*, Annals of University of Petroșani, Electrical Engineering, Universitas Publishing House, Petroșani, Vol.9, pp.312-319, 2007.
- [11] Popescu M.C., Petrișor A., Drighiciu A., *Fuzzy Control Algorithm Implementation using LabWindows-Robot*, WSEAS Transactions on Systems Journal, Issue 1, Vol.8, pp.117-126, January 2009.
- [12] Popescu M.C., Petrisor A., M.A. Drighiciu *Fuzzy Control of the Position for the Piston of an Industrial Robot*. Proceedings of the 12th WSEAS International Conference on Systems, New Aspects of Systems, pp. 222-227, Heraklion, Greece, July 2008.
- [13] Popescu M.C., Borcosi I., Popescu L., Onisifor O., *Simulation Hybrid Fuzzy Control of SCARA Robot*, Journal WSEAS Transactions on Systems and Control, Issue 2, Vol.3, pp.105-114, February 2008.
- [14] Popescu M.C., Borcosi I., Popescu L., Onisifor O., *Simulation Hybrid Fuzzy Control of SCARA Robot*, Proceedings of the 3rd WSEAS International Conference on Applied and Theoretical Mechanics, pp.175-181, Puerto De La Cruz, Tenerife, Canary Islands, December 2007.
- [15] Popescu M.C., Petrișor A., *Neuro-fuzzy control of induction driving*, 6th International Carpathian Control Congress, pp.209-214, Miskolc-Lillafured, Budapest, May 2005.
- [16] Popescu M.C., Manolea Gh., Perescu-Popescu L., Drighiciu A., *Implementation of New Solution Software for Three Tank System Control*, Proceedings of the 11th International Conference on Mathematical Methods and Computational Techniques in Electrical Engineering, Published by WSEAS Press, pp.202-208, Vouliagmeni Beach, Greece, September 2009.
- [17] Precup R.E., *Soluții de conducere fuzzy a sistemelor cu fază neminimă. Aplicații la conducerea hidrogenatoarelor*, Ed. Orizonturi Universitare, Timișoara, 2000.
- [18] Preitl St., Precup R.E., *Introducere în conducerea FUZZY a proceselor*, Ed. Tehnică București, 1997.
- [19] Simulink, *Dynamic System Simulation for Use with Matlab*, User's Guide, MathWorks Inc., Natick, MA, 2004.