Comparison of Simulation Software

EUROSIM - Simulation News Europe features a series on comparisons of simulation software. Based on simple, easily comprehensible models special features of modelling and experimentation within simulation languages, also with respect to an application area, are compared. Features are, for instance: modelling technique, event handling, numerical integration, steady-state calculation, distribution fitting, parameter sweep, output analysis, animation, complex logic strategies, submodels, macros, statistical features etc.

This issue introduces a new comparison C9, see definition and sample solution below. We invite all readers to participate in these comparisons. Please, simulate the model(s) with a tool of your choice and send a report to the editors in the following form (on diskette, any word processing format, or per email or transfer to our ftp-server):

- short description of the language,
- model description (part of source code, diagram, ...),
- results of the tasks with experimentation comments, max. 1 page. (For publication in *EUROSIM - Simulation News Europe* all contributions that exceed one page will be modified by the editors to fit into one page.) Reports of solutions of the Parallel Comparison should not be more than one and a half page in length.

We offer to place the full model (source code, graphics, etc.) and additional information on our WWW server. We also invite you to prepare animations. Please send files and additional information in HTML-format.

The definitions of all comparisons, and an overview on the solutions sent in may be found on our WWW-server: **http://argesim.tuwien.ac.at/comparisons/**

SNE	Comparison									
	C1	C2	C3	C4	C5	C6	C7	CP 1	C8	C9
0	Def									
1	5	Def								
2	4	4	Def							
3	4	3	3	Def						
4	1	5	5	2	Def					
5	4	-	1	1	2					
6	-	2	-	2	1	Def				
7	1	2	1	2	-	1	Def			
8	-	1	-	-	-	1	3			
9	-	-	-	-	-	2	3			
10	1	2	-	-	-	1	2	Def/1		
11	2	2	1	-	1	-	-	2		
12	1	-	1	-	-	-	2	3		
13	-	-	-	-	-		3	1		
14	3	-	1	-	-	-	2	-		
15	-	-	1	-	1	-	-	-		
16	1	-	-	-	-	-	1	-	Def/1	
17	-	-	1	-	1	-	1	1	1	Def/1
Σ	27	21	15	7	6	5	17	8	2	1

Comparison 9: Fuzzy Control of a Two Tank System

The number of applications containing fuzzy components is still increasing. Modern simulation systems provide enhancements to implement fuzzy components in a convenient way.

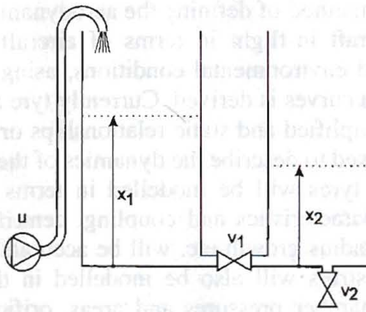

Figure 1: The two tank system

A two tank system (see figure 1) in a specific configuration is characterized by the following nonlinear ODE set

$$f = 0.06624\, v_1 \sqrt{|x_1 - x_2|}\ \mathrm{sign}(x_1 - x_2)$$

$$\dot{x}_1 = 0.067\, u - f$$

$$\dot{x}_2 = f - 0.0605\, r\, v_2\, |x_2|^{0.43}$$

where

$$r = \begin{cases} 1.2 : x_2 < 16 \text{ cm} \\ 1 \ \ : x_2 \geq 16 \text{ cm} \end{cases}$$

This model includes characteristics of the liquid (laminar, turbulent, friction). The valve positions are $v_1 = 0.4$, $v_2 = 0.3$. The task is to control the liquid level x_2. A fuzzy controller is used according to figure 2.

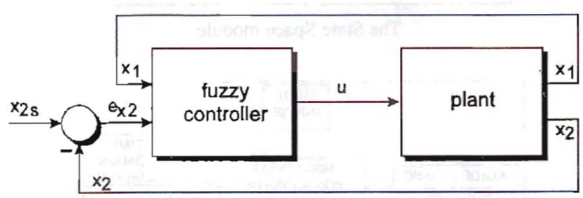

Figure 2: Fuzzy control of liquid level x_2

Two fuzzy controllers FC1 and FC2 will be defined. They should be implemented as discrete systems operating using 1 second sampling time. The membership functions for e_{x2} and x_1 are defined in figure 3. They are the same in the case of FC1 and FC2.

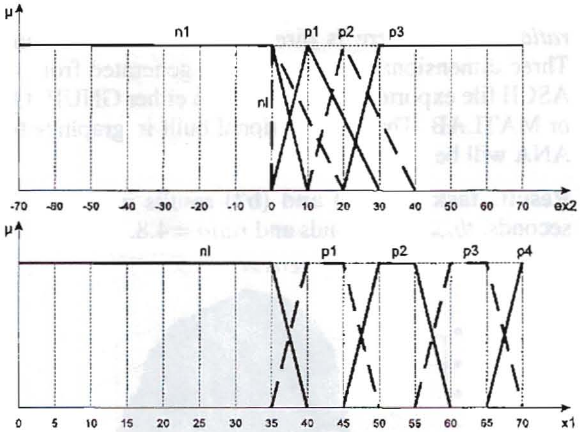

Figure 3: Membership functions of e_{x2} and x_1 for FC1 and FC2

The membership functions for the linguistic output variable u in the case of FC1 are defined in figure 4 and in the case of FC2 singletons are used as shown in figure 5.

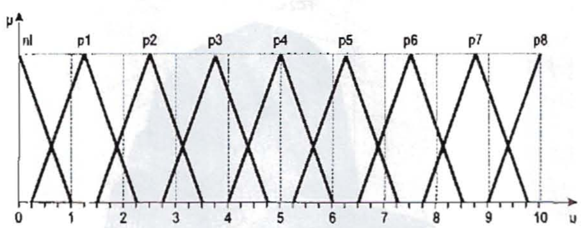

Figure 4: Membership functions of u for FC1

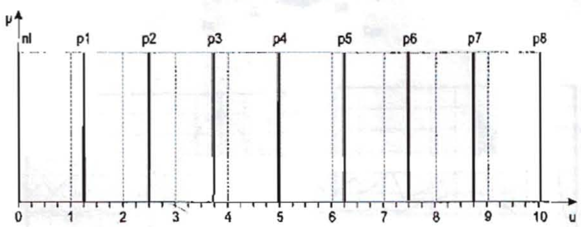

Figure 5: Membership functions of u for FC2

The rulebase (figure 6) should be implemented using the operators *MIN* for AND and *MAX* for OR. For the interference algorithm use *max-prod* and for defuzzyfication *center of gravity*.

		x_1				
		nl	p1	p2	p3	p4
	p3	p8	p7	p5	p3	nl
	p2	p7	p6	p4	p3	nl
ex2	p1	p7	p5	p3	p2	nl
	nl	p4	p3	p2	p1	nl
	n1	nl	nl	nl	nl	nl

Figure 6: Table of rules

The tasks to be performed are:

Task a

- (a1) Describe the features supporting fuzzy control in your simulator or the interface to an interfaced fuzzy tool. Model the controller by means of features of the simulator or an appropriate additional tool linked to the simulator. Give a rough model description of the controller and of the overall model.
- (a2) Compute and visualize the 3 dimensional characteristic (surface) of the fuzzy controller FC1. Place e_{x2} [-70 .. 70] on the x-axis, x_1 [0 .. 70] on the y-axis and u on the z-axis. Subdivide the x and y-axis 40 times (41 times 41 points). State the calculation time ta_{fc1}. Specify the machine used. Since the computation of fuzzy systems is a complex task, the calculation time for 1681 lookups documents the performance of the implementation.
- (a3) Repeat (a2) using FC2. If your system does not support singletons directly, you may use any kind of *emulation*. State the calculation time ta_{fc2} and specify the ratio ta_{fc1} over ta_{fc2}.

Task b

- (b1) Simulate the whole system using FC1 for x_{2s} = 25 cm for 1000 seconds. Plot x_2, x_1 and u versus time. State the computation time over tb_{fc1}.
- (b2) Repeat (b1) using FC2. State the calculation time tb_{fc2} and specify the ratio tb_{fc1} over tb_{fc2}.

Task c

- (c1) FC3 is defined using FAM interference and is obtained by weighting the rules of FC2 according to the following table

		x_1				
		nl	p1	p2	p3	p4
	p3	1	1	0.1	1	1
	p2	1	1	0.1	1	1
ex2	p1	1	1	0.1	1	1
	nl	1	1	0.1	1	1
	n1	1	1	0.1	1	1

Figure 7: Table of weightings

Repeat (a2) using FC3. State the calculation time tc_{fc3} only (no surface plot). Describe how weighting can be implemented into your fuzzy description.
- (c2) Describe the outstanding features of the simulation system regarding to fuzzy modelling.

Johannes W. Goldynia, Johann M. Marinits, Email: goldynia@iert.tuwien.ac.at, marinits@iert.tuwien. ac.at, Vienna University of Technology, Institute for Electrical Control Engineering, Gusshausstr. 27-29/375, A-1040 Vienna/Austria