LA-UR- 01-2805

Approved for public release; distribution is unlimited.

Title: FUZZY CONTROL FOR A NONLINEAR MIMO-LIQUID LEVEL PROBLEM

Author(s): William J. Parkinson, ESA-EPE
Ronald E. Smith, ESA-EPE
Fred N. Mortensen, X-2
Paul J. Wantuck, ESA-EPE

Mohammad Jamshidi, UNM
K. Sean Patrick, UNM

Submitted to: 5th International Conference on Engineering Design and Automation
Las Vegas, Nevada
August 5-8, 2001

Los Alamos
NATIONAL LABORATORY

Fuzzy Control for a Nonlinear-MIMO-Liquid Level Problem

William J. Parkinson[1], Ronald E. Smith[1], Fred N. Mortensen[1], Paul J. Wantuck[1], Mohammad Jamshidi[2], and K. Sean Patrick[2]

Los Alamos National Laboratory, Los Alamos, NM 87544[1]
University of New Mexico Albuquerque, NM 87131[2]

ABSTRACT

Nonlinear systems are very common in the chemical process industries. Control of these systems, particularly multi-variable systems, is extremely difficult. In many chemical plants, because of this difficulty, control is seldom optimal. Quite often, the best control is obtained in the manual mode using experienced operators. Liquid level control is probably one of the most common control problems in a chemical plant. Liquid level is important in heat exchanger control where heat and mass transfer rates can be controlled by the amount of liquid covering the tubes. Distillation columns, mixing tanks, and surge tanks are other examples where liquid level control is very important.
The problem discussed in this paper is based on the simultaneous level control of three tanks connected in series. Each tank holds slightly less than 0.01 m^3 of liquid. All three tanks are connected. Liquid is pumped into the first and the third tanks to maintain their levels. The third tank in the series drains to the system exit. The levels in the first and third tank control the level in the middle tank. The level in the middle tank affects the levels in the two end tanks. Many other chemical plant systems can be controlled in a manner similar to this three-tank system. For example, many distillation column liquid level control problems can be represented as a total condenser with liquid level control, a reboiler with liquid level control, with the interactive column in between. The solution to the three-tank-problem can provide insight into many of the nonlinear control problems in the chemical process industries.
The system was tested using the fuzzy logic controller and a proportional-integral (PI) controller, in both the set-point tracking mode and disturbance rejection mode. The experimental results are discussed and comparisons between fuzzy controller and the standard PI controller are made.

KEYWORDS

Fuzzy Logic, Nonlinear Control; MIMO Control; Liquid Level Control

1. Introduction

In this study we tested five different control systems on our three-tank system depicted in figure 1. One test for each controller was run in the set-point-tracking mode. And six disturbance rejection tests were run for each controller. Only the results from the fuzzy controller and the PI controller tests will be discussed here. And only the results of one of the six disturbance rejection tests will be discussed. Some of the results of the set-point-tracking tests have been presented elsewhere [1**]. This paper is devoted to the disturbance rejection problem. Some of the work presented here was done in the laboratory with the experimental equipment and some work was done with a computer simulation of the experimental system. The computer simulations matched experimental values very closely, so we determined that the simulation results were valid.
The electrical engineering laboratory at the University of New Mexico houses the three-tank system used in this study. It consists of three Lucite tanks in series, each holding slightly less than 0.01 m^3 of liquid. Figure 1 is a schematic diagram of the three-tank system. All three tanks are connected, with the fluid that leaves Tank 2 exiting the system. We pumped liquid into Tanks 1 and 2 to maintain their levels. The fluid levels in these two tanks control the level in Tank 3. The fluid level in Tank 3, in turn, affects the levels in Tanks 1 and 2.

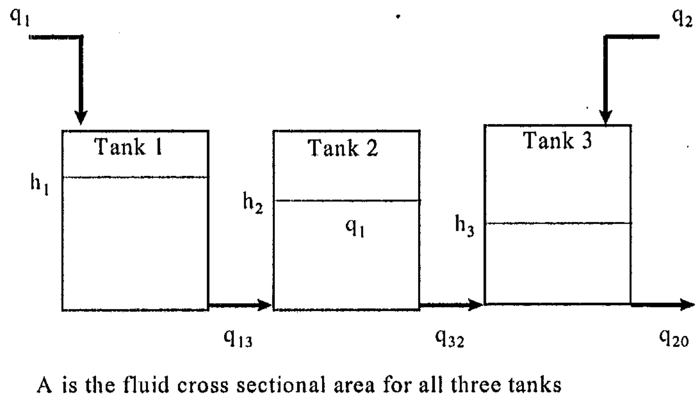

Figure 1. The three-tank system shown with important variables.

2. Mathematical Background

The three differential equations describing the dynamics of this system, using the variable definitions shown in figure 1. are given in equations (1-3).

$$A\frac{dh_1}{dt} = q_1 - q_{13} \tag{1}$$

$$A\frac{dh_3}{dt} = q_{13} - q_{32} \tag{2}$$

$$A\frac{dh_2}{dt} = q_2 + q_{32} - q_{20} \tag{3}$$

The constant A is the cross sectional area of the tanks. The variables h_1, h_2, and h_3 are the current tank level readings for tanks 1, 2, and 3 respectively. The volumetric flow terms q_{13}, q_{20}, and q_{32} are the liquid flows from tank 2 between tanks 1 and 3, and between tanks 3 and 2, respectively. These flows are defined by equations (4-6). The constant g appearing in these equations is the gravitational constant.

$$q_{13} = \Phi_1 \,\mathrm{Sign}(h_1 - h_3)\sqrt{2g\,|h_1 - h_3|} \tag{4}$$

$$q_{20} = \Phi_2\sqrt{2gh_2} \tag{5}$$

$$q_{32} = \Phi_3 A_n \,\mathrm{Sign}(h_3 - h_2)\sqrt{2g\,|h_3 - h_2|} \tag{6}$$

The constants Φ_1, Φ_2, and Φ_3 are friction flow coefficients for the valves and pipes between the tanks, multiplied by the cross sectional flow area of the pipes.

Each tank in the three-tank system is supplied with a valve used to simulate a leak. The valves between the tanks were used to simulate partial plugs. Since we were actually dealing with valves and not real leaks or plugs, we could set the valves partially open (or partially closed). These partially opened valves would have valve or friction coefficients, ϕ_i. For the disturbance rejection tests we used coefficients that would cause significant upsets, but that wouldn't overwhelm the system. In practice these coefficients would not be known because we would get random leaks and plugs, but they were used in the simulation model. Each test was run for 1000 seconds. Between 200 and 700 seconds the disturbance was introduced. After 700 seconds the disturbance was removed. This actually made for two tests in one. It required the controllers to make one adjustment and then adjust back to normal conditions

3. The PI Controller

The proportional-integral (PI) controller is the classical control system that we compared with the fuzzy technique in this study. The PI controller output, or control action, is defined by equation (7).

$$u_i = K_p e_i + K_I \int_0^T e_i dt \tag{7}$$

The variable e_i is the difference between the set-point level and the current level measurement $w_i - h_i$, or the error. The terms K_p and K_I are the proportional and integral control constants, respectively. The integral upper limit, T, is the current time, measured from the set-point change at time = 0. For the PI control case equation (7) describes the control actions u_1 and u_2, that control the flow from the pumps q_1 and q_2, depicted in figure 1. A great deal of time was spent optimizing the control constants, K_p and K_I. This was done with the set point tracking problem in mind. The PI controller performed nearly as well as the fuzzy controller in the set point tracking mode. But not nearly as well in the disturbance rejection mode.

4. The Fuzzy Controller

Of the five control systems, the fuzzy controller was the easiest to design. Our basic idea was to put the controller designer in the shoes of a skilled plant operator, and ask the following question:
"How would I design the system if I were to operate it manually?"
We assumed that if the operator were filling the tanks, he/she would turn the pumps to maximum flow, until the level was a few centimeters from the set-point level. Then the flow could be adjusted toward the set-point flow, until it reached the set point. This control system required only six rules, three for each pump. The rules are the same for each pump and they are given in Table 1. They are of the form:
If Error(i) is ... Then Flow_Change(i) is ...

Table 1. Fuzzy Rules for Set Point Tracking, for i = both 1 and 2

Rule Number	If (Error(i)) is	Then (Flow_Change(i)) is
1	Error(i) = Negative	Flow_Change(i) = Negative
2	Error(i) = Zero	Flow_Change(i) = Zero
3	Error(i) = Positive	Flow_Change(i) = Positive

We have defined the term Error(i) as follows for both Tanks 1 and 2.

$$\text{Error(i)} = \frac{w_i - h_i}{Rangeh(i)} \text{ , for i = 1 or 2} \tag{8}$$

If $w_i > h_i$, then Rangeh(i) equals w_i. Otherwise Rangeh(i) equals $h_{iMax} - w_i$, for i = 1 or 2.
The variable h_i is the current level for tank i and w_i is the set point for tank i. The term h_{iMax} is the maximum level for tank i, or the level when the tank is full. The Flow_Change(i) variable is defined by equation (9).

$$\text{Flow\_Change(i)} = \frac{q_i - q_{iss}}{Rangeq(i)} \tag{9}$$

If Flow_Change(i) > 0 then Rangeq(i) equals q_{iss}. Otherwise, Rangeq(i) equals $q_{iMax} - q_{iss}$, for i = 1 or 2. The variable q_i is the current pump flow, from pump i and q_{iss} is the steady state, or set point, flow for pump i. This term is computed from a simple algebraic mass balance for each new set point level, w_i. The term q_{iMax} is the maximum possible flow from pump i.

Figures 2 and 3 are the input and output membership functions that are connected by the rules shown in Table 1.

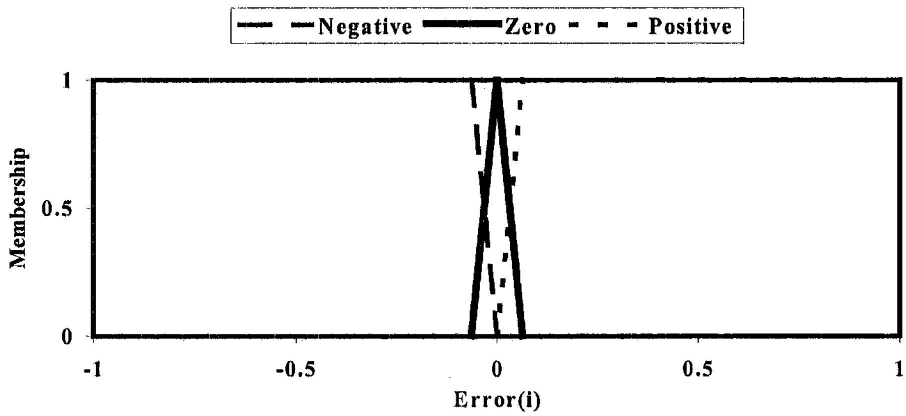

Figure 2. Input membership functions for Errror(i) for i = both 1 and 2.

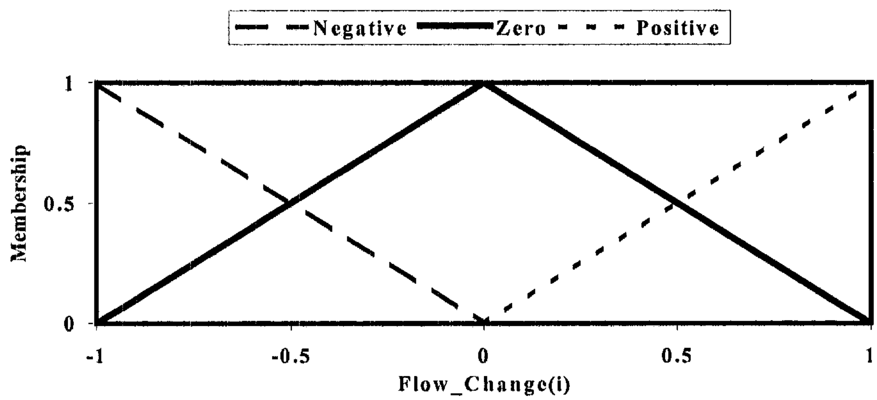

Figure 3. Output membership functions for Flow_Change(i) for i = both 1 and 2.

5. The Fuzzy Logic Controller and Disturbance Rejection.

The fuzzy logic controller is actually a hierarchical controller, with two sets of rules. One set of rules covers set point tracking, as described above. The other set covers disturbance rejection. The hierarchical system has two parts. They are the decision making part and the fuzzy rules for disturbance rejection. The same fuzzy rules are used for both parts. The input membership functions are slightly different when the system is trying to decide if it should be in disturbance rejection mode, than when it is actually in that mode. The rules determine if the controller should be in the disturbance rejection mode and also the size of what we call a ***help factor***.

In the disturbance rejection mode we have lost our calculated material balance, since we either have a leak or a plug. These conditions were not in the original material balance equations used to compute the steady state pump flows, q_{iss} in equation (9). So we must find our new steady state pump flows, q_{iss}, by using a root finding technique.

There are 18 rules, nine for tank 1 and pump 1, and nine for tank 2 and pump 2. The rules are exactly the same for each tank. The rules are of the form:

If Error(i) is … and If Slope(i) is ...Then Help_Factor(i) is …

Where Error(i) is defined by equation (8), Slope(i) is defined by equation (10), and Help_Factor(i) is a fuzzy factor used to speed the convergence of the root finding algorithm.

$$Slope(i) = \frac{h_i^{j+1} - h_i^j}{\Delta t} \tag{10}$$

Where Δt is the sample interval equal to time t(j+1) minus time t(j), and h_i^{j+1} and h_i^j are the level measurements of tank i at times t(j+1) and t(j), respectively.
The input membership functions Error(i) and Slope(i) are each represented by three simple triangles much like those shown in figure 3. The output membership functions for Help_Factor(i) are also represented by simple triangles, but in the case of the output there are five membership functions. If Slope(i) has the wrong sign relative to Error(i), it is detected by the control system and interpreted as a disturbance rejection condition. A root finding technique is then used to find a new q_{iss} to be used with the rules in Table 1 and the membership functions shown in figures 2 and 3. The root finding techniques is just the simple iterative method defined by equation (11) defined in most numerical methods texts. For example see Henrici [***42].

$$x^{j+1} = f(x^j) \qquad (11)$$

Where j is the iteration number and the time between iterations is the sample interval. This technique is quite stable but also quite slow. The fuzzy ***help factor*** speeds it up significantly. This is also an approximation of the human approach to solving the disturbance rejection problem.

6. The Experiemnt

Six disturbance rejection tests were performed for each controller. The results from test #1, a leak in tank 1 with a friction coefficient of 0.65, are shown in figure 4 and 5 for the fuzzy controller and PI controller respectively.

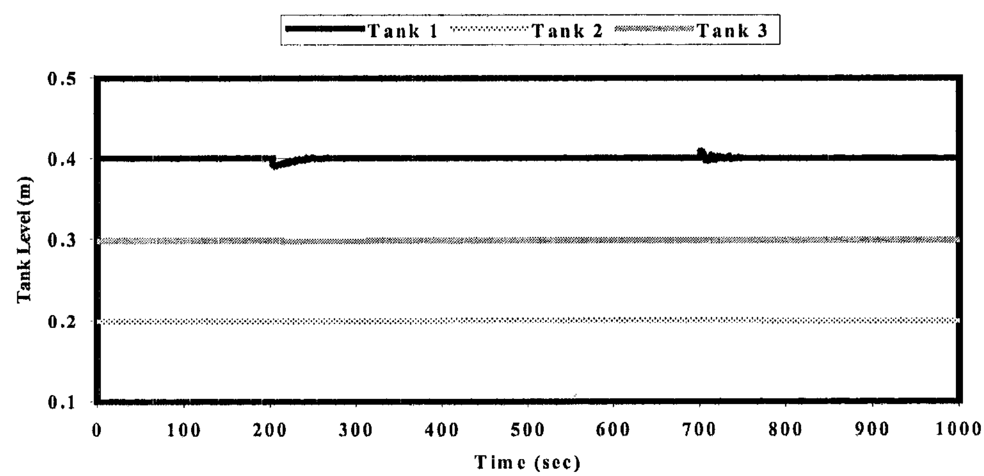

Figure 4. Fuzzy disturbance rejection for a leak in tank 1.

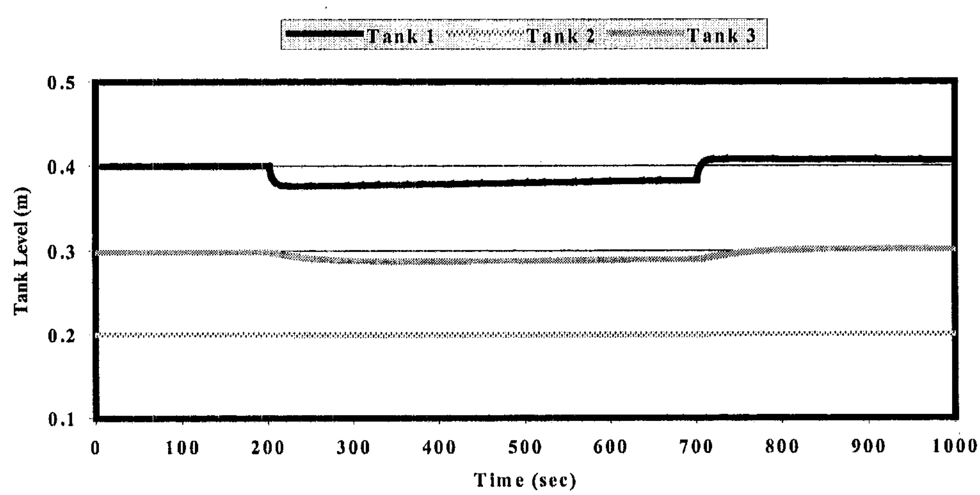

Figure 5. PI controller, disturbance rejection for a leak in tank 1.

Figures 4 and 5 show that fuzzy logic controller handles the disturbance rejection for a leak in tank 1 quite well, and the PI controller not as well. Numerical results presenting the integral of the errors, the difference between the tank level and the set pint, are shown in figure 2, for this test. The controllers handled the other tests in a similar manner.

Table 2. Integral errors for test number 1 for the fuzzy logic and PI controllers.

Controller and Tank	Total Integral Error	Positive Integral Error	Negative Integral Error
Fuzzy Logic controller/ Tank 1 -- level	0.3933	0.2880	0.1053
Tank 2 -- level	0.0035	0.0020	0.0015
Tank 3 -- level	0.1455	0.1165	0.0290
PI controller/ Tank 1 -- level	12.1279	10.0535	2.0744
Tank 2 -- level	0.3293	0.2494	0.0799
Tank 3 -- level	5.6128	4.9072	0.7056

7. Comparison of the Controllers.

With the set point tracking problem, described elsewhere [1**], the fuzzy controller took first place with the PI controller, with control constants optimized for the set point tracking problem was a close second. The major problem with all but the fuzzy controller, is the time consuming effort to optimize, or find the best, controller constants. The fuzzy controller only requires setting up the rules and membership functions correctly. The input membership functions for these problems were designed by knowing the maximum pump flow rates, the diameter of the tanks, and a proper sample time.

For the disturbance rejection problem, the fuzzy logic controller again took first place in the production of the smallest error. The PI controller described here placed last. It is likely that with a different set of control constants, K_p and K_I, The PI controller would do better in the disturbance rejection mode, but it is hard to tune a controller for an unexpected disturbance.

The results of this study indicate that the fuzzy logic controllers are the best ***overall*** controllers for this type of control problem.

REFERENCES

1. --------- *Laboratory Setup --Three-Tank-System, DTS200 Instruction Manual.* amira GmbH. Duisburg, Germany (1997). – Original article (not seen) Freund E.u. and H. Hoyer. "Das Prinzip der nichtlinearen Systemenkopplung mit der Anwendung auf Industrieroboter."*Zeitschrift: "Regelungstechnik"*,Nr. 28 im Jahr 1980, S.80-87 und S. 116-126.
2. Ogunnaike B.E., and W. H. Ray *Process Dynamics, Modeling, and Control.* Oxford University Press, New York (1994).