Proceedings of the 2003 Winter Simulation Conference
S. Chick, P. J. Sánchez, D. Ferrin, and D. J. Morrice, eds.

INCORPORATING FUZZY LOGIC ADMISSION CONTROL IN SIMULATION MODELS

Qisheng Le
Gerald M. Knapp

The Industrial and Manufacturing Systems Engineering Department
The Louisiana State University
Baton Rouge, LA 70803, U.S.A.

ABSTRACT

Admission Control is of great interest in computer, communication and production network applications. As systems become more complex or need to make decision based on multiple objectives or require to satisfy several constraints, Fuzzy Logic Control (FLC) is a natural tool to handle admission problems. This paper discusses the framework of a tool developed for easily integrating fuzzy logic admission control into Arena simulation models. The tool is implemented as a "drop-in" model block that performs admission control. The block can be configured to perform simple or priority-based admission, as well as multiple queue selection. The FLC "rules" are set through a design-time user interface and stored in an MS Access database. Because multiple FLC blocks may be included in a model, almost any possible admission scenario (multiple entry points, parallel lines, etc) may be modeled. A case study is used to demonstrate the developed framework.

1 INTRODUCTION

Control of arrivals to queuing systems is an important aspect of the design of computer, communication and production network applications. Fuzzy Logic Control (FLC) has been successfully applied to many types of system admission control problems. In the admission problem, the controller acts as a gate, making decisions on whether a new arrival will be allowed to enter a system or subsystem, and in the case of parallel paths, which path it is to be admitted to. The objective is to improve the systems performance, such as maximizing the average profit, minimizing the average cost, increasing the utilization of the resources, or satisfying quality of service metrics. For example, Zhang and Phillis (2001) examined the use of fuzzy logic to solve the admission control problem in two simple series parallel networks; the first case has one workstation with two parallel ones and the second has two parallel servers and three arrival processes by independent Poisson streams. Zhang and Phillis (1999a, 1999b) also proposed a framework using fuzzy logic to handle the arrivals control to tandem queues with two stations and to assign dynamically to idle servers in queuing systems with heterogeneous servers. Fuzzy Logic has also been applied to admission control in communication networks (Ko et al. 2001; Lim et al. 2001a, 2001b; Mehrvar et al. 1997).

This paper discusses the framework of a tool that has been developed for easily integrating fuzzy logic admission control into Arena simulation models. The tool is implemented as a "drop-in" model block that performs admission control. The framework can handle quite general admission control issues. The system can include as many queues as needed and each queue may have its own constraints such as finite buffers. The FLC block is independent of the arrival process, and so any inter-arrival distribution may be used. Servers may have different services rates and functions.

The paper is organized as the following: the proposed framework and its implementation procedures are presented in the next section. Section 3 describes a case study including the example problem, FLC settings and implementation, and the results of two example experiments. Finally, conclusions and further research directions are discussed in the Section 4.

2 FLC FRAMEWORK

Figure 1 illustrates the basic idea of the proposed framework. The framework consists of three components: the simulation modeling environment, the FLC block component, and a supporting database.

The FLC block is implemented in Arena 7.0 as an ActiveX component. Arena provides a Visual Basic programming interface for integration with ActiveX components.

The user places the FLC block in the simulation model any place they wish to control an admission process. The block has a design time interface (basically properties pages) that provide the ability to customize each FLC blocks operation.

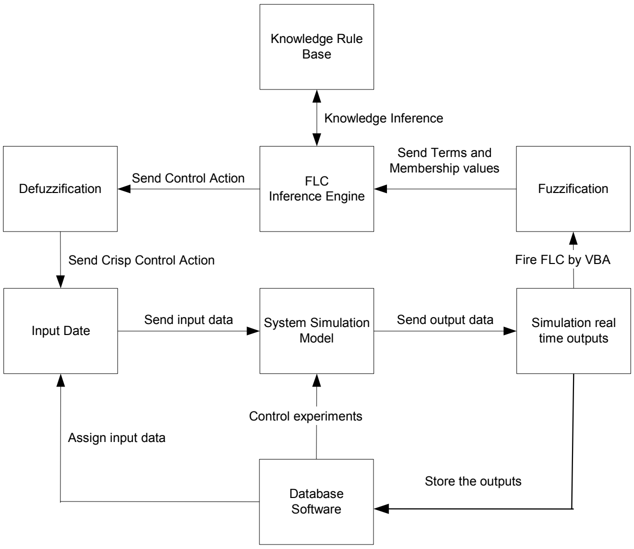


Figure 1: Framework

The FLC component stores the blocks custom information in a Microsoft Access database, keyed by a unique identifier for the block. This information includes:

- What input parameter(s) will be used for the decision process. Parameters are selected from Arena system attributes, entity attributes, and/or Arena runtime statistics (multiple may be selected).
- Whether different rules are to be applied for different priority levels, and if so, what entity attribute provides the priority level.
- Whether the block must decide between multiple output paths, and if so, what their block labels are.
- For each priority level and each input parameter, the set of fuzzy terms and their membership functions. The membership functions are specified graphically, similar to Matlab's fuzzy set toolbox.
- The target and improvement direction for the FLC input parameters.
- The fuzzy rules under which admission will either be granted or denied (the system completes any unspecified combinations).
- Block label where to send rejected entities.
- The fuzzy controller type (discussed below).
- The user has the option of using previous saved blocks as templates, saving on data entry for similar FLC cases.

When an entity enters the FLC block in the simulated system, it triggers the FLC decision process. The basic logic of the FLC is as follows:

- The FLC code gathers the current values of its input parameters by querying the Arena simulation engine.
- The input parameters are "fuzzified" according to the specified fuzzy linguistic terms and their membership functions defined for the FLC block.
- Rule consequences are then computed and aggregated to a fuzzy set "control action" based on the type of fuzzy controller selected, such as the Mamdani or Sugeno Controller (Zimmermann 1999).
- The fuzzy set is then "defuzzified" since a crisp control action is required. The FLC block makes a decision to accept or reject the incoming entity based on the defuzzification result.
- The entering entity is routed to the appropriate block label.

3 CASE STUDY

The developed framework is tested using a case study to dynamically control the arrivals to a series queuing systems based on the selected real time simulation outputs, as shown in Figure 2. It consists of 8 stages. Entities are assumed to go through one or more but up to 8 stages. In this paper, we assume that every entity will be served by the same stages if accepted. A gate is set up at the entrance of the system to control the admission of new incoming entities. The gate is denoted by a Branch block and controlled by the FLC block. If accept = 1, then accept the incoming entity; if accept = 0, then reject it.

In the main interface, we can enter information about the replication, inter-arrivals, and the number of servers and the service rate for each stage. In Arena model, it is easy to set dispatching rule and the capacity of the queue, the schedule policy of servers, and other nonlinearly quantitative parameters.

The simulation outputs Work in Process (WIP), the number of entities in each queue (NQs) are selected as the input variables of FLC. If the system has only one stage, then the controlled variables are WIP and NQ1 (the number of the entities in Queue 1); If the system has 6 stages, then the controlled variables are WIP, NQ1, NQ2, NQ3, NQ4, NQ5, and NQ6 (the number of the entities in Queue 1, 2, 3, 4, 5, and 6). Triangular fuzzy sets are used due to computational efficiency (Zimmermann 1999), as shown in Figure 3. The language terms are defined as the error between the ideal and actual value of each FLC input variable. They are Positive Big (PB), Positive Medium (PM), Positive Small (PS), and Zero (ZO).The membership value "error" of each input variable, e, is defined by

$$e = (h - Setpoint) * ScaleFactor \qquad (1)$$

where h denotes the actual value for WIP and NQs; *ScaleFactor* is equal to 6 if the membership function is shown as in Figure 3; *Setpoint* P_0 denotes the ideal value of each FLC input variable.

P_0 and $P_{Allowed}$ can be decided by historical data, or simulation output without FLC, practical constraints, experts concerned, or any combinations above.

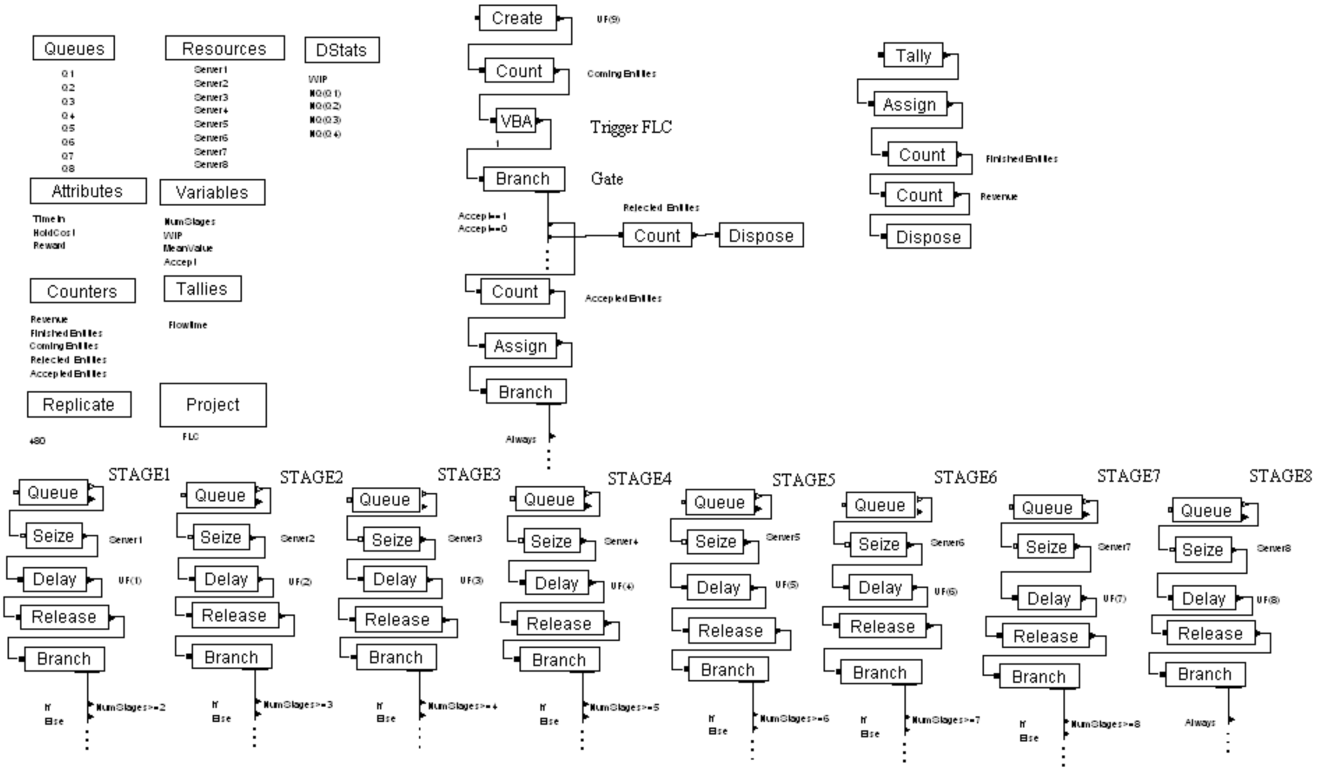
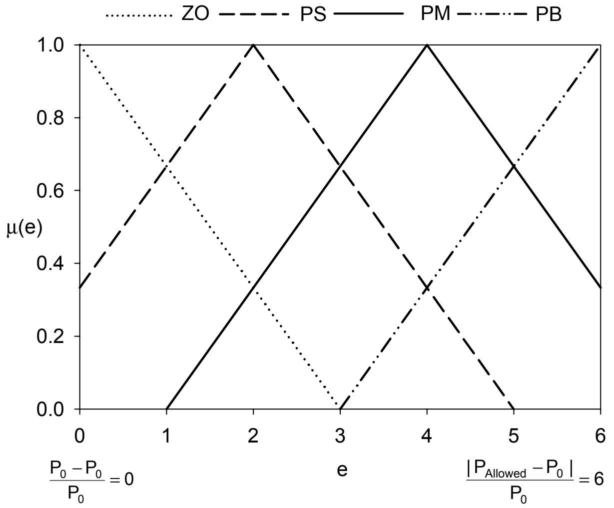


Figure 2: A Simulated System



Note: $P_{Allowed}$ is the allowed maximum or minimum value of individual input variable.

Figure 3: Membership Function of Input Variables WIP and NQs

The output of FLC is a crisp value, D_1, to indicate to accept or reject the incoming entity. Thus D_1 is either 1 or 0. The membership function of D_1 is given in Figure 4.

The Mandani inference rule (Zimmermann 1999) is employed in this research. The rules connect the input variables with the output variables. They are based on the

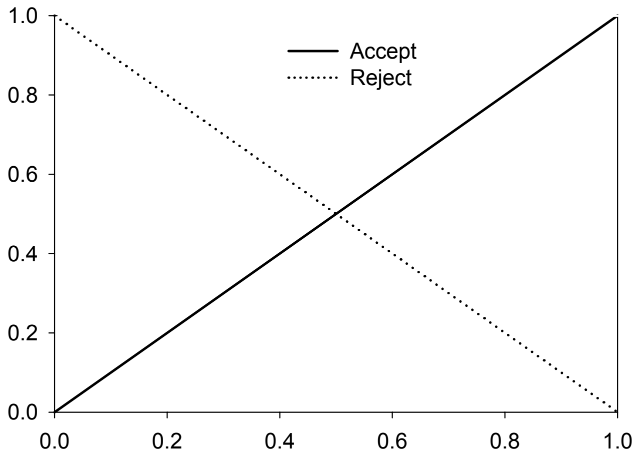


Figure 4: Membership Function Output Variable (D1)

fuzzy state description that is obtained by the definition of linguistic variables. Formally, the rules in this example can be written as

Rule r: if x_1 is $A_1^{j_1}$ and x_2 is $A_2^{j_2}$ and…and x_n is $A_n^{j_n}$,
then $D_1 = 1$ or $D_1=0$.

Where x_1, x_2, … are the term values of input variables; $A_1^{j_1}$, $A_2^{j_2}$, … are the term values in the rule base to lead to one control action.

The rule base for control action is determined by the actual control objectives. In this paper, we assume the following rules to reject the incoming entity and all of the other rules to accept the entity, which will be implemented in the VBA code.

1. Any one of term values of all input variables is Positive Big (PB).
2. Any two term values of all input variables are Positive Medium (PM) or Positive Big (PB).

The height method for defuzzication (Zhang and Phillis 1999a, 1999b, 2001) is used in this research. The expression is as follows:

$$D_1 \approx \frac{\sum_{i=1}^{m} e^{(i)} f_i}{\sum_{i=1}^{m} f_i} \tag{2}$$

where D_1 denotes decision value, the rounding integer which is the closest to the result of right hand side; m denotes the total number of implications; $e^{(i)}$ equals 0 or 1 acording to reject or accept the incoming entity; f_i denotes the membership value of the i-th implication. If D_1 is 1, then accept the incoming entity into the system; if D_1 is 0, then reject the entity.

We consider the admission control for two cases: single stage and 6 stages. We implemented the experiments on the PC with 400MHz Pentium II Processor and 128 MB system RAM running Arena 7.0 professional edition. Both cases ran 20 replications with 480 minutes in each replication. The input data, FLC information and the experiment results (directly available from the database) are listed in the Tables 1-8. From the trace results in the MS Access, we can also view all records regarding the real time values of the FLC input variables and final decision to accept or reject the incoming entity. The experiment time for single stage case is about 10 seconds; the experiment time for 6 stages case is about 14 minutes.

Table 1: Simulation Input Data for Single Stage System

Simulation input parameter	Value
Replication run length	480
Number of replications	20
System arrival distribution	Exponential (7)
Number of stages	1

Table 2: Stage Information for Single Stage System

Stage No.	Number of Parallel Servers	Service Time of Each Server
1	4	Exponential (50)

Table 3: Limit and Target Values of FLC Input Variables for Single Stage System

Input variable	Limit $P_{Allowed}$	Target P_0, or Setpoint
WIP	15	5
NQ(1)	10	1

Table 4: Some Statistics Results for Single Stage System

Simulation output	Average	Standard deviation	Minimum	Maximum
Coming Entities	70.5	8.34	56	89
Accepted Entities	47.35	6.52	36	62
Rejected Entities	23.15	9.66	8	52
Finished Entities	37.35	7.04	26	52
Flow Time	4.55	0.697	3.25	5.68

Table 5: Simulation Input Data for 6 Stage System

Simulation input parameter	Value
Replication run length	480
Number of replications	20
System arrival distribution	Exponential (3)
Number of stages	6

Table 6: Stage Information for 6 Stage System

Stage No.	Number of Parallel Servers	Service Time of each server
1	4	Exponential (25)
2	1	Exponential (5)
3	2	Exponential (15)
4	1	Exponential (6)
5	3	Normal (20,3)
6	2	Lognormal (13,2)

Table 7: Limit and Target Values of FLC Input Variables for 6 Stage System

Input variable of FLC	Limit $P_{Allowed}$	Target P_0, or Setpoint
WIP	50	30
NQ(1)	10	1
NQ(2)	10	1
NQ(3)	10	1
NQ(4)	10	1
NQ(5)	10	1
NQ(6)	10	1

Table 8: Some Statistics Results for 6 Stage System

Simulation output	Average	Standard deviation	Mini-mum	Maxi-mum
Coming Entities	161.8	14.61	138	192
Accepted Entities	71.65	6.07	58	82
Rejected Entities	90.15	16.62	64	128
Finished Entities	46.05	4.97	34	54
Flow Time	3.64	1.2	1.68	5.59

4 CONCLUSIONS AND FUTURE STUDY

A framework for admission control to real world queuing systems is proposed by the means of the integration of FLC, database management and simulation technology. The queuing systems may include multiple objectives and constraints. A case study has shown the framework to be easy to use, and having the flexibility to deal with complex systems. We are currently designing a more user-friendly interface to enter data for the FLC.

REFERENCES

Ko, Y.-C., S.-C. Park , C.-Y. Chun, H.-W. Lee; and C.-H Cho. 2001. An adaptive QoS provisioning distributed call admission control using Fuzzy logic control. *IEEE International Conference on Communications*, v2, p356-360.

Lim, H.H., and B. Qiu. 2001a. A predictive measurement-based fuzzy logic connection admission control. *IEEE International Conference on Communications*, v3, p920-924.

Lim, H. H., and B. Qiu. 2001b. Fuzzy logic traffic control in broadband communication networks. *IEEE International Conference on Fuzzy Systems*, v1, p99-102.

Mehrvar, H.R., and T. Le-Ngoc. 1997. Fuzzy logic in estimation of traffic burstiness for admission control in broadband networks. *IEEE International Conference on Communications*, v 2, p1090-1094.

Zhang, R., and Y. A. Phillis. 2001. Admission control and scheduling in simple series parallel networks using fuzzy logic. *IEEE Transactions on Fuzzy Systems*, v9, n2, April, p307-314.

Zhang, R., and Y. A. Phillis. 1999a, Fuzzy control of arrivals to a tandem queueing system. *IEEE Trans on Fuzzy Systems*, Vol. 7. No. 3, p361-167.

Zhang, R., and Y. A. Phillis. 1999b. Fuzzy control of queueing systems with heterogeneous servers. *IEEE Trans. Fuzzy Systems*, vol.7, no.1, p17-26.

Zimmermann, H. J. 1999. *Fuzzy Set Theory and its Applications*. 2nd edition. Kluwer Academic Publishers.

AUTHOR BIOGRAPHIES

QISHENG LE is a PhD student in the Department of Industrial and Manufacturing Systems Engineering at Louisiana State University (LSU). He has a B.S. degree in Mechanical Engineering from Northeast University, China and M.S. degrees in Mechanical Engineering from Central South University of Technology, China. Before studying at LSU, he worked in mechanical engineering for 10 years. His research interests include the applications of simulation and information technology and artificial intelligence to system operations, and distributed simulations. His email address is `<qle1@lsu.edu>`.

GERALD M. KNAPP, PhD, PE, is Fred B. & Ruth B. Zigler Associate Professor of the Engineering in Industrial & Manufacturing Systems Engineering at Louisiana State university. He has B.S. and M.S. degrees in Industrial Engineering from SUNY Buffalo, and a Ph.D. in Industrial Engineering from the University of Iowa. His research specializations include enterprise / information systems architecture and design, and reliability engineering. His email address is `<gknapp@lsu.edu>`.