Expert Systems with Applications 38 (2011) 4860–4865

Contents lists available at ScienceDirect

Expert Systems with Applications

journal homepage: www.elsevier.com/locate/eswa

A fuzzy logic based multi-agents controller

E.A. Olajubu *, O.A. Ajayi, G.A. Aderounmu

Department of Computer Science and Engineering, Obafemi Awolowo University, Ile-Ife, Nigeria

ARTICLE INFO

Keywords:
ABD
ANG
MASs
MASC
Fuzzy logic
Controller

ABSTRACT

This paper presents a fuzzy logic based controller (Multi-Agents System Controller (MASC)) which regulates the number of agents released to the network on a Multi-Agents Systems (MASs). A fuzzy logic (FL) model for the controller is as presented. The controller is a two-inputs-one-output system. The controllability is based on the network size (NTZ) and the available bandwidth (ABD) which are the inputs to the controller, the controller's output is number of agents (ANG). The model was simulated using SIMULINK software. The simulation result is presented and it shows that ABD is the major constraint for the number of agents released to the network.

1. Introduction

The phenomenal growth and usage of ubiquitous computer network is fueling researches on economic, easy and secured means of connecting to geographically distributed data either in an intranet or on the Internet. The mobile agent technology has been seen as a technology that offers the omnipotent solution to this quest of computer network end-users (Kotz & Gray, 1999). Mobile agents are programs that can autonomously migrate from node to node in a networked environment to perform some computational tasks on behalf of the user. The running code suspends it execution and it is transported to another node or place depending on the destination, where it resume execution from where it stopped. Mobile agents are well known technology which is very effective for implementing complex software and network-centric applications such as e-commerce, telecommunication, Information retrieval systems, process control, etc. The technology offers reduce network latency; encourage optimal bandwidth usage which is be limited by several technical factors. Many end-users (mostly the developing nations' end-users) still get connected by means of modem, or at most, ADSL over the traditional copper wire. Many other end-users will connect via low-bandwidth wireless networks who cannot afford to have more than 128 Kbps to 1 Mbps available at their desktop or palmtop (Amitava, 1999). The infrastructure and network resources limitations make mobile agent a de facto technology for Internet and network-centric applications. It provides a distinct, common framework in which geographically dispersed, information-oriented applications can be implemented efficiently and easily, with the programming burden spread evenly across information, middleware, and client providers. In other words, mobile agent gives network providers the time and flexibility to provide their users with *more* useful applications, each with more useful features.

Mobile agent technology has gained a wide range of acceptance among Computer Science researchers and network-centric applications developers due to some salient advantages the technology promise to offer over the traditional client-server architecture. The wide acceptance of the technology is due to its fault-tolerance during network downtime and reduced latency (bandwidth optimum usage) on computer network (Kotz & Gray, 1999). The advantages and potential of agent-to-agent communication has greatly fuel Multi-Agents Systems (MASs) researches, therefore MASs has been seen as a tool to solve many complex Internet applications (Jennings, 2001; Wooldridge, 2002). The authors of Abrahams and Dai (2005) and Espinasse, Fournier, and Freitas, 2008 proposed MASs architecture for information storage, search and retrieval on the web. In the same vain (Mabry, Eetters, Schneringer, Edwards, & Hug, 2003; Sumit, Qutaiba, Jerry, & Aivars, 1998; Webber & Silva, 2008) aimed at solving complex problems of distributed diagnosis, patients monitoring over the Internet through MASs. According to (Mandureira et al., 2008), the appropriate technology for removing the complexity in network scheduling, especially in manufacturing companies' computer network is through MASs applications. The proposal to solve complex network applications through MASs by researchers has increased greatly with little or no considerations of how the MASs will on bandwidth during their itinerancy. It is clear that, mobile agents are program codes that make use of bandwidth during their itinerancy on computer network. The amount of bandwidth usage is a function of the number of agents in the systems (Baek et al., 2001a, 2001b) and our simulation experiment support this view. To strike a balance between the numbers of agents in a MASs and the available bandwidth this paper propose

* Corresponding author.
E-mail addresses: emmolajubu@oauife.edu.ng (E.A. Olajubu), anuajayi@oauife.edu.ng (O.A. Ajayi), gaderoun@oauife.edu.ng (G.A. Aderounmu).

0957-4174/$ - see front matter © 2010 Elsevier Ltd. All rights reserved.
doi:10.1016/j.eswa.2010.09.034

a flexible fuzzy logic based multi-agent controller that will regulate the number of agents release to the network based on available bandwidth. The rest of the paper is organize in the following way, Section 2 elucidates some FL controllers, Section 3 describes our model in FL environment, Section 4 discusses our simulation environment and simulation results while Section 5 gives the conclusion of the paper.

2. Fuzzy logic controller applications

Fuzzy logic (FL) models have superseded traditional and convectional know-hows in science and engineering applications, especially in the field of control engineering. The wide acceptance of FL is due to its simplicity, flexibility and ability to mimic human knowledge that is difficult to computerize in traditional way. There are many proposals of FL applications in literatures, some of them are FL controllers which have been fully developed into useful applications; the approach of the development (software or hardware) is a function suitability of implementation. FL controllers have been proposed by researchers for a number of systems such as communication networks (Lin, Wong, & Dillon, 2005; Sumit et al., 1998), robot navigation (Ono, Uchiyama, & Potter, 2004; Wang & Lui, 2008), food processing (Odejobi, 2007; Odejobi & Owolarafe, 2005), Diagnosis (Hooshmand & Banejad, 2006; Soliman & Rizzoni, 1997), Manufacturing (Azadeh, Anvari, & Izadbakhsh, 2007; Lau, Cheng, Lee, & Ho, 2008; Tsourveloudis & Phillis, 1998), convenient stores (Leung, Lau, & Kwong, 2003), etc. FL based control system have become very popular in this era of technology. Some conventional applications are in use such as rotary cement kilns (Tayel, Rizk, & Hagras, 1997), an air conditioning system (Wichasilp, Wiriyasuttiwong, & Kantapait, 2003). Also, there are fuzzy logic based applications for household appliances' control for steady use like cameras and camcorders, and the control of dishwashers (Badami & Stephanou, 1997) and similar appliances. It is clear, that fuzzy logic principles have been applied in various industrial sectors which have been developed to full fledge applications that are in use in many human endeavor. While these areas have fully benefited from fuzzy logic system development, the regulation of mobile agents in multi-agent systems is yet to gain this advantage. This issues is address in this work so as to minimize bandwidth wastage during mobile agents routing in multi-agent system. The capability to formalize approximation of the reasoning analysis offered by fuzzy logic provides a promising solution to issues of bandwidth and the number of agents routing a network for computational task. Therefore, since there is yet be any agent controller (either using conventional method or fuzzy logic approach) that regulates the number of agents in MASs to the best of our knowledge, this paper stands to fill this gap by proposing a fuzzy logic based approach to regulating the number of agents released to a network in a multi-agent system, based on the network size (NTZ) and the available bandwidth (ABD).

3. Model development in fuzzy logic

There are two inputs to the controller: (i) the number of nodes on the network (NTZ) and (ii) available bandwidth (ABD) while the number of Agents (NAG) to be deployed to the network is the out output of the controller. The two inputs determine the amount of agents to be generated for a particular network size.

3.1. Data fuzzification

Data fuzzification is the first process in modeling a fuzzy logic system. The first step in the modeling of the controller is data fragmentation (fuzzification) into input that can be accepted by fuzzy logic. The fuzzification converts each unit of input data to a degree of membership by a call on some membership function in the fuzzy logic toolbox. In the process of fuzzification, each input data is mapped with the conditions of the rules to establish the degree of fitness on how each rule matches the particular input. The data fuzzification table is presented in Table 1.

The rule-based system for the controller uses two input variables and one output variable as both the conditions and the conclusion of the rules. The multi-input–single-output (MISO) applies to this controller. The input variables for the controller are network size (NTZ) and available bandwidth (ABD), while the single output is the number of agents (NAG).

3.2. Rule format

The rule format for the controller is of the form "if. . ..then" format. There are 56 rules in the knowledge base of the controller. The "if. . ..then" rule statement is used to formulate the conditional statements that consist the knowledge base. It assumes the form "if *Y* is *k* then *Z* is l" the if part is called the premise whereas the then part is called the consequent. The rule base of this system

Table 1
Data fuzzification.

FLV	Linguistic Implication	NTZ	FLV	Linguistic Implication	ABD
1a: Network size			*1b: Available bandwidth*		
VS	Very small network	20–75	VP	Very poor Bandwidth	0.1–1.0
ST	Small network	50–150	PR	Poor Bandwidth	0.8–1.5
AT	Average network	100–250	FR	Narrow Bandwidth	1.1–1.3
FL	Fairly large network	180–350	GD	Fair Bandwidth	2.5–4.5
LT	Large network	300–450	LR	Good Bandwidth	3.5–6.0
VL	Very large network	400–600	VL	Large Bandwidth	5.0–7.0
EL	Extra large network	530–750	MX	Very Large Bandwidth	6.5–8.5
IT	Increased network size	680–880	MB	Maximum Bandwidth	8.0–10.0
MT	Maximum network	820–1000			
c: Number of agents					
FLV	Linguistic implication	NAG			
VA	Very small agents	1–3			
SA	Small number of agents	2–5			
AV	Average number of agents	4–8			
FA	Fairly large number of agents	6–10			
LA	Large number of agents	9–13			
IA	Increased large number of agents	11–15			
LV	Very large number of agents	14–18			
MA	Maximum number of agents	17–20			

makes use of forward chaining system. The forward chaining system process the initial fact first, the rules are used to draw conclusion based on the processed data. The forward chaining system is said to be data driven. Samples of the applied rules are shown below.

- IF NTZ IS ST AND ABD IS PR THEN NAG IS VA
- IF NTZ IS AT AND ABD IS FR THEN NAG IS SA
-]IF NTZ IS FL AND ABD IS MB THEN NAG IS LV

3.2.1. Membership function

Membership function assigns membership grades level to each element in a fuzzy set. The membership function decides the degree of membership of each element. The transition from member to non-member is gradual rather than swift. In this work the function $\mu_{(x)}$ is used to assign membership grade to every element in the fuzzy set. A certain degree of overlap is allowed in the design of the membership function so that the controller may have a well define output. The membership function for the input and output were symmetrical triangles. The inputs NTZ and ABD membership function are depicted in Fig. 1a and b respectively and the membership function for the output NAG is presented in Fig. 1c as shown below.

3.2.2. Controller surface plot

The two inputs and one output has input-output mapping as shown in Fig. 2. This is the mesh plot of the relationship between the two inputs (NTZ and ABD) and the output NAG. The plot comes out from the rule base which contains 56 rules and the surface of the plot is more or less bumpy. The conical surface of the plot is due to the triangular shape of the inputs. The vertical shape from the origin shows the sensitivity of the controller to change in any of the inputs. This is an advantage because the controller is very sensitive therefore it can easily respond or peak a little change in

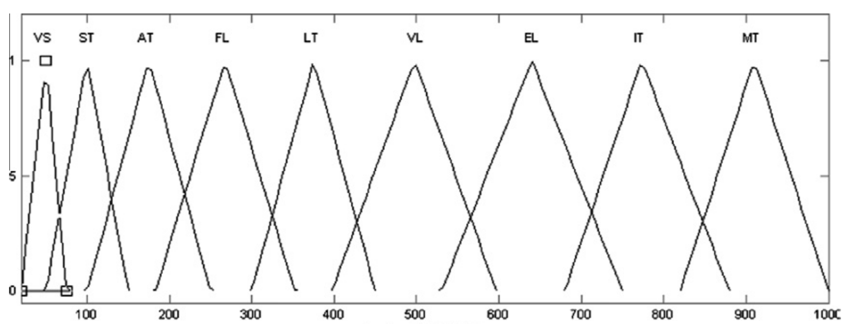

Fig. 1a. Network size membership function (generated from Matlab fuzzy logic toolbox).

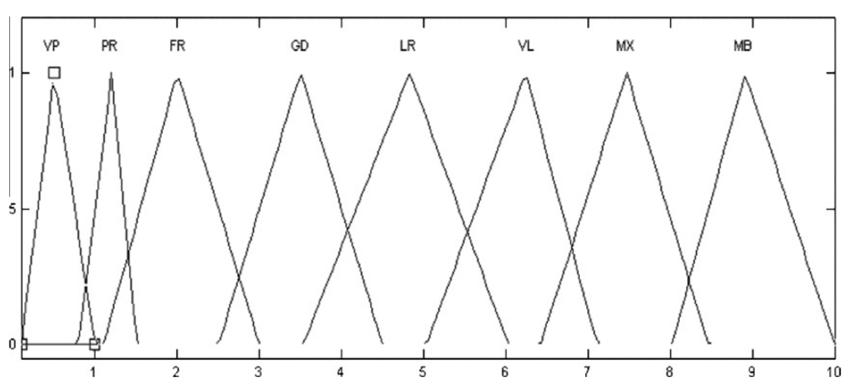

Fig. 1b. Available Bandwidth Membership function (generated from Matlab fuzzy logic toolbox).

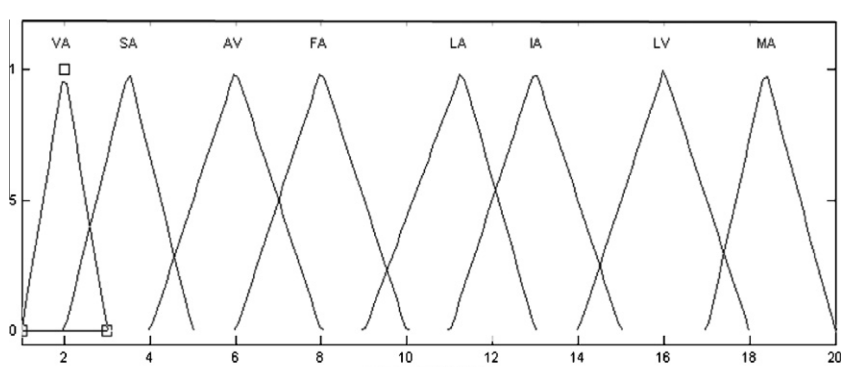

Fig. 1c. Number of agents membership function (generated from Matlab fuzzy logic toolbox).

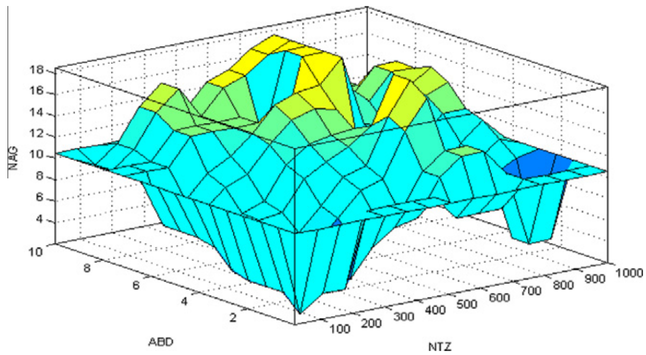

Fig. 2. Controller surface viewer.

either of the inputs. The depressions at the top of the mesh plot signifies large network with narrow bandwidth while the conical structures implies large network with appreciable bandwidth. It is noteworthy, that ABD plays a significant roll in allocation of agents to any network. If the NTZ is large with a narrow ABD, definitely the agents' allocations are going to be few compared to the NTZ. If the ABD is proportional to NTZ, agents' allocation to the network will be more to secure a high response time.

4. Overview of simulation environment

MATLAB is fast becoming a popular model development tool both in industry and learning institutions for science and engineering applications. It is a tool suitable for building system models by researchers and academia in various fields especially in technology. It is a technical high-performance language for computing with many instructions and apparatus for designing applications and evolving computer algorithms. Tools like SIMULINK provides a good GUI and block diagram module that allows users to easily build, simulate and test system models at the same time through standard and custom block libraries. In this research, the fuzzy logic model is developed in MATLAB fuzzy logic toolbox while the developed model was simulated in SIMULINK. The intelligent system determines the number of agents allocated to a network based on the size of the network and available bandwidth.

4.1. Description of the simulation parameters

The simulated experiment allocates agents to a network in a multi-agent system based on the available bandwidth (ABD) and network size (NTZ). Three experiments were conducted for the simulation. We conducted our simulation on a Pentium 2.8 GHz CPU and 512 MB of RAM running on the Windows XP operating system. Fig. 3 depicts the simulation model based on the model described in Section 3. The two inputs uses random number generator to determine the inputs to be fed in to the system during simulation, the random number generator is a function provided in SIMULINK environment in which this work was simulated.

4.2. MASC model development

The MASC is the organ that regulates the number of agents that released to a network (depending on the size of network). The designed model is composed of two sources empowered with random number generator. The two sources are the two inputs NTZ and ABD to the system. The input NTZ has random number generator with the lower band 100 and upper band of 1000 nodes for the simulation while ABD random number generator with lower band 0 and upper band of 1200 KB/s. The two inputs were linked together and connected to the fuzzy logic controller developed in the MATLAB fuzzy logic toolbox called controller which is imported to SIMULINK environment for simulation purpose. Fig. 4 depicts the system model used for the MASC controllability during system simulation. The fuzzy logic Controller in Fig. 3 has all the defined rules discussed in Section 3. The fuzzy logic controller is linked to the scope which is the output of the system. The scope is the number of agents (ANG) required on the network based on the available bandwidth (ABD). The simulation in this section

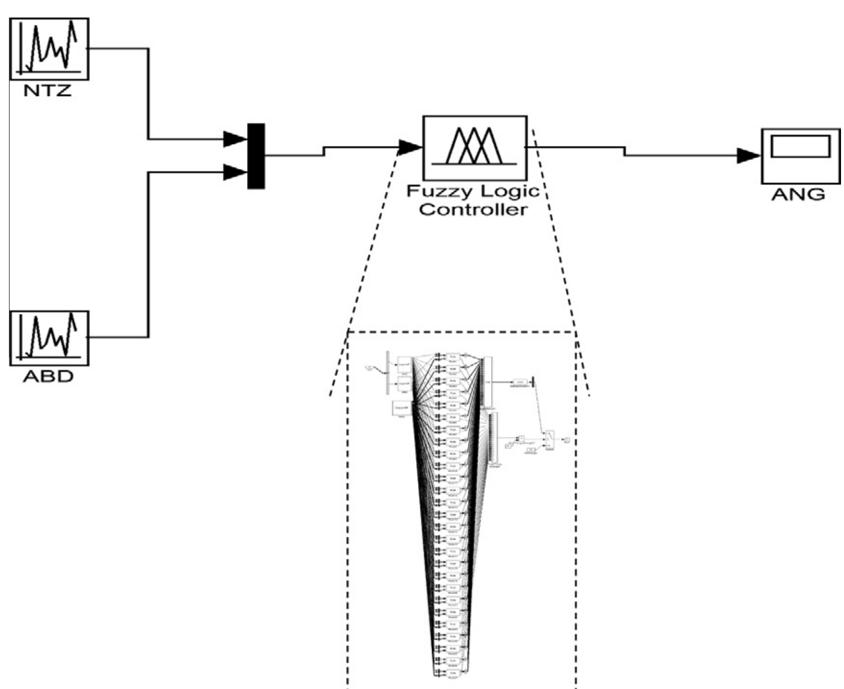

Fig. 3. Controller model.

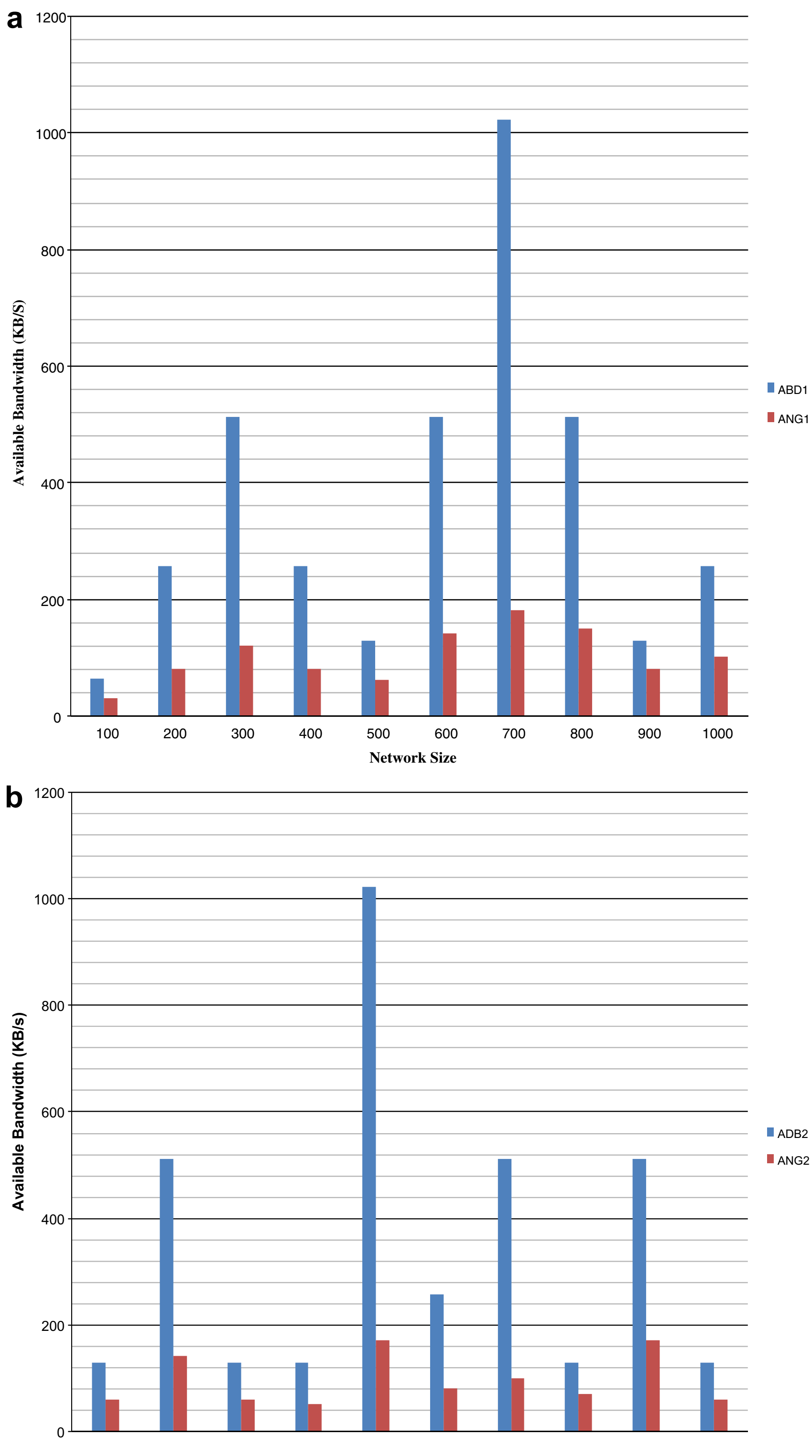

Fig. 4. (a) The result of Experiment 1. (b) The result of Experiment 2. Simulation experimental results.

discusses the controllability of the designed MASC using fuzzy logic design system.

4.3. Controller simulation result

Three different simulation experiments were conducted for this work. The system run for some time so as to gain stability then the readings were taken. The results of the experiment were consistent as shown in Fig. 5a–c for the three experiments. The graphs show the number of agents released to a particular network size of certain bandwidth. The available bandwidth (ABD) on the network plays a very important role in the determination of the number of agents that will be appropriate for a network of certain size. A network with large number of nodes but with a narrow bandwidth are apportioned few number of agents that will not constitute unnecessary burden on the narrow network bandwidth, while a network of the same size with larger bandwidth has larger number of agents. The obvious difference on the two networks is the response time. The response time of large network with narrow bandwidth is going to be low while the response time of the same network with larger bandwidth is going to be higher. When the agents that route large network with narrow bandwidth are few, definitely, the response time will be very low. If the agents on the network with large bandwidth are many, the response time will be very high. It will be expected to have increase in agents going to the network as the network size increases. This is not so because of available bandwidth factor. For example, from the simulation result, about fifteen agents were released to a network of a hundred computers, while about 17 agents were released to a network of four hundred computers. The increase in the growth of network size does not correspond with the increase in the number of agents that were sent to the network. This is based on the bandwidth available to each of the network. Two runs of simulation experiments were done and results were depicted in Fig. 4a and b. The input ABD has ABD1 and ABD2 which are the variable inputs for the two experiments. Also, the output ANG has ANG1 and ANG2 which are the experimental results (outputs) for the three experiments. The number implies the first, second and third experiment respectively.

5. Conclusion

A fuzzy logic based controller model is developed and simulated for a MASs. The controller promised to minimize bandwidth wastage due to excess agents routing communication network for given task. The developed system is very sensitive to available bandwidth, if the bandwidth is wide; likely the agent allocation to the network will be high subject also, to the number of computers or nodes on the network. The fuzzy logic capability to formalize approximate reasoning analysis is explored in this work to create a sensitive controller for MASs. If the model is implemented in real-life situation, it will reduce over flooding computer network with mobile agents in MASs.

Acknowledgment

We are sincerely grateful to Academic Staff Union of Universities (ASUU) for their support for this work through ASUU RESEARCH GRANT AWARD OF 2007. It should be noted that the views expressed in this paper are that of the authors and not ASUU.

References

Abrahams, B., & Dai, W. (2005). Architecture for automated annotation and ontology based querying of semantic web resources. In *Proceedings of the 2005 IEEE/WIC/ACM international conference on Web intelligence* (pp. 413–417).

Amitava, D.-R. (1999). Bringing home the Internet. *IEEE Spectrum, 36*(3), 32–38.

Azadeh, M. A., Anvari, M., & Izadbakhsh, H. (2007). An integrated FDEA-PCA method as decision making model and computer simulation for system optimization. In *Proceedings of the 2007 summer computer simulation conference* (pp. 609–616).

Badami, V. V., & Stephanou, H. (1997). Fuzzy logic-based supervisory control of household appliances. In *Proceedings of the sixth IEEE international conference on fuzzy systems* (Vol. 1 (1–5), pp. 213–218).

Baek, J., Kim, J., & Yeom, H. (2001). Timed mobile agent planning for distributed information retrieval. In *Proceedings of the fifth international conference on Autonomous agents* (pp. 120–121).

Baek, J., Kim, G., Yeo, J., & Yeom, H. (2001). Cost effective mobile agent planning for distributed information retrieval. In *Proceedings of the 21st international conference on distributed computing systems* (pp. 65–72).

Espinasse, B., Fournier, S., & Freitas, F. (2008). Agents and ontology based information gathering on restricted Web domain with AGATHE. In *Proceedings of the 2008 ACM symposium on applied computing* (pp. 2381–2386).

Hooshmand, R., & Banejad, M. (2006). Application of fuzzy logic in fault diagnosis in transformers using dissolved gas based on different standards. In: *Proceedings of World academy of science, engineering and technology* (Vol. 17, pp. 151–161).

Jennings, N. R. (2001). An agent-based approach for building complex software systems. *Communications of the ACM, 44*(4), 35–41.

Kotz, D., & Gray, R. S. (1999). Mobile agents and the future of Internet. In *ACM operating system review* (pp. 11–12).

Lau, H. C. W., Cheng, E. N. M., Lee, C. K. M., & Ho, G. T. S. (2008). A fuzzy logic approach to forecast energy consumption change in a manufacturing system. *Expert Systems with Applications, 34*(3), 1813–1824.

Leung, R. W. K., Lau, H. C. W., & Kwong, C. K. (2003). On a responsive replenishment system: A fuzzy logic approach. *Expert Systems, 20*(1), 20–32.

Lin, W. W. K., Wong, A. K. Y., & Dillon, T. S. (2005). A novel fuzzy logic controller (FLC) for shortening the TCP channel roundtrip time by eliminating user buffer overflow adaptively. In *Proceedings of the 28th Australasian conference on computer science* (Vol. 38, pp. 29–37).

Mabry, S., Eetters, T., Schneringer, T., Edwards, N., & Hug, C. (2003). Addressing the complexity of patient monitoring with multiagents and modular logic. In *Proceeding of the second international joint conference on autonomous agents and multiagents systems* (pp. 1058–1059).

Mandureira, A., Santos, F., & Preveria, I. (2008). *Self managing agents for dynamic scheduling in manufacturing. Paper presented at GECO proceedings of the 2008 GECO conference Companion on Genetic and Evolutionary Computation, Atlanata, USA* (pp. 2187–2192) New York: ACM.

Odejobi, O. A. (2007). Computational modeling of systems engineering: A case study of the cassava processing plant. *Journal of Computer Science & Its Applications, 14*(2), 1–9.

Odejobi, A. O., & Owolarafe, O. K. (2005). Computer simulation of fuzzy control system for Gari fermentation plant. *Journal of Food Engineering, 68*, 197–207.

Ono, Y., Uchiyama, H., & Potter, W. (2004). A mobile robot for corridor navigation: a multi-agent approach. In *Proceedings of the 42nd annual southeast regional conference* (pp. 379–384).

Soliman, A. A., & Rizzoni, G. R. (1997). *The application of fuzzy logic to the diagnosis of automotive systems, 1997*. Available at: http://www.sae.org/technical/papers/970208.

Sumit, G., Qutaiba, R. H., Jerry, S., & Aivars, C. (1998). A survey of recent advances in fuzzy logic in telecommunications networks and new challenges. *IEEE Transactions on Fuzzy Systems, 6*(3), 1998.

Tayel, M., Rizk, M. R. M., & Hagras, H. A. (1997). A fuzzy logic controller for a dry rotary cement kiln. *IEEE Fuzzy Systems, 3*(1), 1525–1531.

Tsourveloudis, N. C., & Phillis, Y. A. (1998). Manufacturing flexibility measurement: A fuzzy logic framework. *IEEE Transactions on Robotics and Automation, 14*, 513–524.

Wang, M., & Lui, J. N. K. (2008). Fuzzy logic-based real-time robot navigation in unknown environment with dead ends. *Robotics and Autonomous Systems, 56*(7), 625–643.

Webber, C. G., Silva, J. L.T. (2008). Self and nonself distrimination agents. In *Proceedings of the 2008 ACM symposium on applied computing* (pp. 1987–1988).

Wichasilp, C., Wiriyasuttiwong, W., & Kantapait, K. (2003). Design of fuzzy logic controllers by fuzzy c-means clustering. *Thammasat J. Sc. Tech, 8*(2), 12–16.

Wooldridge, M. (2002). *An introduction to multiagent systems*. Indianapolis (IN): John Wiley & Sons, Ltd.