International Journal of Electrical and Computer Engineering (IJECE)
Vol. 9, No. 2, April 2019, pp. 1201~1208
ISSN: 2088-8708, DOI: 10.11591/ijece.v9i2.pp1201-1208

Elastic neural network method for load prediction in cloud computing grid

Kefaya S. Qaddoum[1], Nameer N. El Emam[2], Mosleh. A Abualhaj[3]
[1]Department of Computer Information Systems, Higher Colleges of Technology, United Arab Emirates
[2]Department of Computer Science, Philadelphia University, United State
[3]Department of Information Technology, Amman Al Ahliyya University, Yordania

Article Info

Article history:

Received Oct 10, 2017
Revised Sep 21, 2018
Accepted Oct 11, 2018

Keywords:

Cloud computing
Load balancing
Neural networks
Virtual machine migration

ABSTRACT

Cloud computing still has no standard definition, yet it is concerned with Internet or network on-demand delivery of resources and services. It has gained much popularity in last few years due to rapid growth in technology and the Internet. Many issues yet to be tackled within cloud computing technical challenges, such as Virtual Machine migration, server association, fault tolerance, scalability, and availability. The most we are concerned with in this research is balancing servers load; the way of spreading the load between various nodes exists in any distributed systems that help to utilize resource and job response time, enhance scalability, and user satisfaction. Load rebalancing algorithm with dynamic resource allocation is presented to adapt with changing needs of a cloud environment. This research presents a modified elastic adaptive neural network (EANN) with modified adaptive smoothing errors, to build an evolving system to predict Virtual Machine load. To evaluate the proposed balancing method, we conducted a series of simulation studies using cloud simulator and made comparisons with previously suggested approaches in the previous work. The experimental results show that suggested method betters present approaches significantly and all these approaches.

Corresponding Author:

Kefaya Qaddoum,
Department of Computer Information Systems,
Higher Colleges of Technology,
Traffic Department Road, Al Ain, United Arab Emirates.
Email: K.s.Qaddoum@gmail.com

1. INTRODUCTION

According to the National Institute of Standards and Technology (NIST): "Cloud computing is a model for enabling convenient, on-demand network access to a shared pool of configurable computing resources [1]. Cloud is a huge scale distributed computing, transferring computer operations and storage from desktop and portable PCs to a huge hosting center [1], where resources could be easily used remotely, with effective cost [2], [3]. Data centers use Virtual Machine (VM) for integration of different solutions [4]. Infrastructure-as-a-Service (IaaS), Platform-as-a-Service (PaaS), and Software-a-a-Service (SaaS) are the offered types of market services among cloud.

Many millions of online devices are connected through the Internet, where using cloud computing service industries is increasing every day [4]. Cloud computing involves sharing supplies, software, which supposed to reduced functioning cost, regarding the time needed to process data, adjust the system permanence according to demands changes in the system as it goes. Load balancers are part of cloud structure; its goal is to forward incoming request to the customer who demands the service. Depending on availability and existing load, the load balancer task is to apply scheduling algorithm to decide the best server that could handle the request [5]. It gets an update on client's server's status and current job. If one server is

overloaded with tasks, then it is redirected to another server to keep load balanced [6-7]. In this paper, the objective of load scheduling is minimizing the average shift time of a set of online loads, where shifting time represents the total elapsed time for a load application from load submission to load completion, including waiting time and execution time.

As elaborated in [8], load computing is a good way to understand job parallelism where each job requires a single processor for execution and different tasks might run concurrently or follow superiority restrictions. On the other side, parallel programs are written with some message-passing library, e.g., MPI [9], are a classic way to accomplish data parallelism.

A load model can be used to describe the scheduling job to keep balance among the cloud VMs. As the high-performance computing platform evolves into grid and cloud environments, the underlying speed-heterogeneous multi-cluster design [10] makes mixed-parallel load scheduling more laborious and difficult than traditional task- parallel loads scheduling because of the resource division issue [11] experienced by parallel task distribution. In [12], Online Load Management was proposed as the first step to tackling such problems. ANN is widely used in different aspects and applications [17], in this work, we propose an Elastic Artificial Neural Network (EANN) framework, which formalizes the online-load scheduling process into job allocation. The job-clustering phase manages the job load within a server. The job rearrangement phase allows some jobs reallocations to different servers to handle the request to increase resource utilization on other servers. The job allocation phase assigns a proper set of resources to a job.

We live in the cloud era, where users of the cloud are increasing fast, that caused heavy job on cloud servers which made the load balancing as an urgent issue of cloud computing. The core goal of load balancing is to allocate the job consistently to the whole cloud to guarantee no overloaded or underloaded servers exists in the entire network [13]. Load balancing algorithms' goal is to monitor the nodes for underloading or overloading and take appropriate action to normalize the load among VMs. So that utilization of resources is better. Many load controlling techniques, like ASKALON [14], DAGman [15], have developed systems to manage load uses in both parallel and distributed systems. Some existing load management systems, e.g., DAGman and [16], make sure the execution of a load does not violate the precedence constraints among tasks but pay little attention to improving scheduling time. Other systems focus on completing task scheduling within loads to enhance desired final time, e.g. [17].

The rest of the paper covers. Section 2 discusses related work of load balancing; Section 3 discusses the proposed system, including Load balancing, suggested framework, algorithm to predict cloud load, server's classification used in Cloud computing. Section 4 shows simulation and results of the suggested system. Section 5 has the conclusion with some ideas for future work.

2. RESEARCH METHODOLOGY

Explaining research chronological, including research design, research procedure (in the form of algorithms, Pseudocode or other), how to test and data acquisition [18], [19]. The description of the course of research should be supported references so that the explanation can be accepted scientifically [20], [21].

Whenever there is a certain job on the Cloud, the load balancer sends the job request through the suggested algorithm to be clustered to which server to be sent [22]. K-means clustering is one of the simplest clustering algorithms used for unsupervised learning problems, MacQueen, 1967 introduced it, it works mainly on minimizing the distance between the cluster center and the data point. K-means is used in this research appointing jobs as data points on the server and based on the processor type and speed to cluster these jobs. After determining hard disk capacity, another clustering is done based on their cost per hour. An operator node to manage node performance exists, its job is to supervise the node then report utilization and send it to manager controller, the manager is a cluster management system aimed at decreasing the effort done to manage cluster resources while refining the facility of customer's demand and manage resources. The manager waits until the whole cluster is empty before looking for work from other clusters. Figure 1 [7] shows system model, where the manager controller sends a report as input which is used in prediction algorithm.

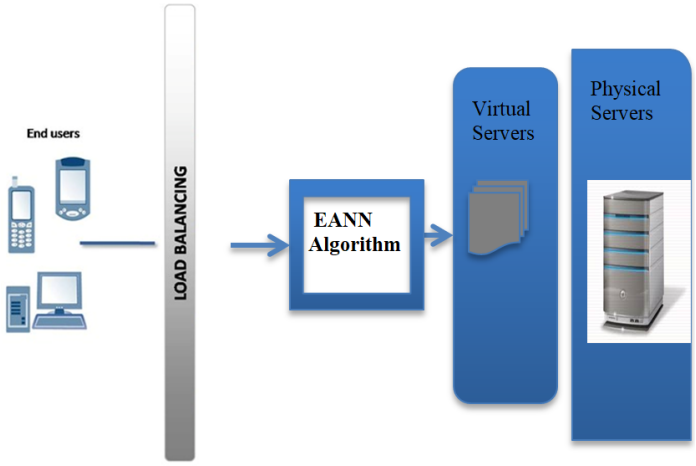

Figure 1. Load balancer

2.1. EANN design for load prediction

To forecast expected resource requests, we need to look inside a VM and analyze logs of pending requests. To do this, VMs modifications are required, which might be difficult. The prediction is built on the historical log behaviors of VMs. The algorithm deploys an artificial neural network to predict load balancing in the cloud; this is achieved by training neural network on a big dataset containing different load situations. Back Propagation learning algorithm was used during EANN training phase so that it can manage incoming jobs to the cloud. The suggested EANN consist of three tiers; the first tier is input, presenting the current job for N nodes. The second is the hidden tier, while the third is output tier, which signifies the balanced job for N nodes. Each node in the input tier act for either the current server's job or the current average job of a cluster of servers where its given a number. On the other hand, the equivalent node in the output tier represents either server's job or cluster's regular job after balancing correspondingly.

The system proposed is an intelligent technique, where the values of the neural network connection weights are modified within training phase, using a reformed optimization methodology; it consists of building an elastic artificial neural network (EANN) based on modified adaptive smoothing errors (MASE). Then to adjust the neural network connection weights during the training phase. The algorithm uses EANN technique work on reducing errors.

First, we need to determine average with weights by a predetermined scheme, where the expected and detected load at a given period *p* as seen in (1).

$$D(p) = c^* D(p - 1) + (1 - c) * O(p); 0 < c < 1 \quad (1)$$

as *c* indicates, the balance among steadiness and approachability.

EANN is used to predict the load of CPU on a defined server. We group the time c =0.5 and use it to predict the next minute load and compare it with the current minute.

The next step is monitoring decreased usage; we need to work on the observed values which are higher by 69% normally.

If balance values groups within [0 to 1], then the balance is within the desired range. We give c -1 rate, so that value we get from equation (1) is converted as (2) here.

$$D(p) = - |c|^* D(p - 1) + (1 + |c|) * O(p); -1 < c < 0 = O(p) + |c| * (O(p) - D(p - 1^*c)) \quad (2)$$

The predicted value is concluded based on historical behaviors on the cloud.

2.2. EANN architecture

The suggested system includes three tiers, neurons in the first, second, and third tiers. EANN as shown in Figure 2 goes as follows:

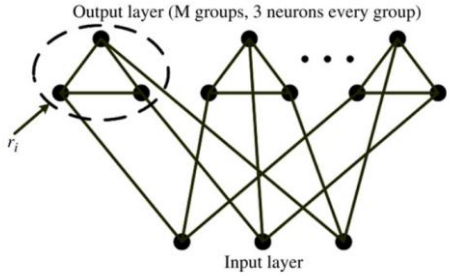

Figure 2. Elastic ANN architecture

Phase 1: Regroup neuron and allocate neurons with diverse routes locations.

Phase 2: group a location route Xs of server node n as an input. Since there are three neurons in the input tier, Xs should be the input of different neuron of input tiers each time.

Phase 3: Create a group B relates server node s corresponds to the neuron n.

$$S = \{i|Xi \in Sq \cap kXs - Xik \leq ri\} \quad (3)$$

where *Xi* is the location route of neuron n and the associations related to *Sq* can shield server node n.

Phase 4: If group S is unfilled, iterate this procedure by picking another random server node.

$$Xn* = \min kXs - Xnk. \quad (4)$$

$$n \in Sq$$

Phase 5: alter the chosen sites of neuron and next ones according to:

$$Xn* = Xn* + \alpha(Xs - Xn) \quad (5)$$

Else, go back to step (2).

The mean squared error (MSE) between the predicted and the required result represents the fitness of the input.

$$\Phi(x) = \frac{2}{1+\exp(1-x)} - 1 \quad (6)$$

$$\delta_z = (t_z - O_z)\frac{d(\Phi(oz))}{do} \quad (7)$$

$$\Delta W_{tz}^{new} = \beta\alpha_{tz}^{new}\delta_z H_t + (1-\beta)\Delta W_{tz}^{old} \quad (8)$$

The factor δ_z and the activation h_t of the hidden neuron H_t, determines adjustment made to the value of V_{st} and error factors be outlined in (8).

3. RESULTS AND ANALYSIS

In this section, it is explained the results of research and at the same time is given the comprehensive discussion. The experiments were conducted on a cloud simulator. In cloud simulator, the cloudsim, that aids modeling, and simulation of Cloud computing systems and application settings. It provisions both system and behavior modeling of Cloud system components such as data centers, virtual machines (VMs) and resource provisioning policies. Using the load balancing parameters of virtual machine and data center, we have examined on cloudsim-3.0 a cloud-computing simulator and evaluate the performance of the suggested method using job assigning to the cloud subset. Figure 4 represents the load balancing in each cloud partition, and the proposed approaches such as EANN based load balancing algorithm shows improved results than the existing techniques. In the beginning simulator first, generates the input workload consisting of a set of workflows arriving at different time instants in an online manner.

For each task of a workflow, the number of processors required to execute it is set by picking up a random number within a pre-configured range according to a specified probability distribution.

Each experiment invokes 20 runs, of which each simulates 100 online workflows in a multi-cluster environment composed of 5 clusters each containing 50-70 processors. EANN algorithm is used to give best possible allocation results. Set p=16 s for example and a set of demands on service is D, where clustered sets represented by C1, C2, C3. The outcome of 92 iterations was balanced and reasonable node allocation. Task allocation using EANN goes as Association C1={S7, S15, S20}, that is, servers S7, S15, and S15 are elected to track demand D1. Coalition C2={S2, S6, S22}, that is, servers S2, S6, and S22 are elected to track demand D2. Coalition C3={S8, S17, S18}, that is, servers S8, S17, and S18 are elected to track demand D3. Results are shown in Table 1.

Table 1. Time of Execution of the Selected Customer's Cloud Tasks

Set-up	Retort	Processing time
First	Data center with 30 VMs	244.68
Second	Two data centers with 15 VMs each	260.77
Third	Two data centers with 30 VMs each	174.92

The predicted value, which is done with the help of past input value is clearly shown in Figure 3. The predicted value provides a higher efficiency of output. Since the predicted value provides greater efficiency, the mean square error is slightly high. As Figure 3 illustrates output if the rate=0.7, and this is selected on account of least error calculation for our jobs prediction on VM, where this result enhances the prediction accuracy and work for the benefit of better scheduling in the next step.

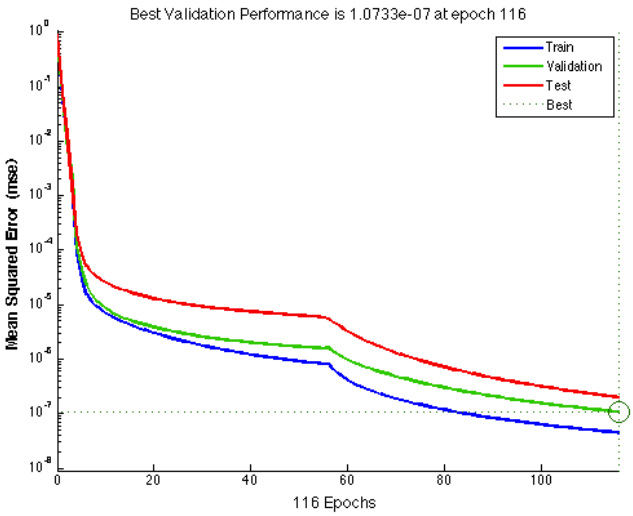

Figure 3. Error rate for best performance

The load migration before and after is given in Figure 4, as we can see that over a period the number of jobs executed comparing to available VMs is becoming reasonable after migrating jobs around. The result of predicted jobs is keeping the balance of different servers as equal as possible. Figure 5 gives the graphical demonstration of the load balancing in each cloud partition and the suggested approach EANN based load balancing algorithm shows the better results than the other techniques like Ant colony and Honey behavior.

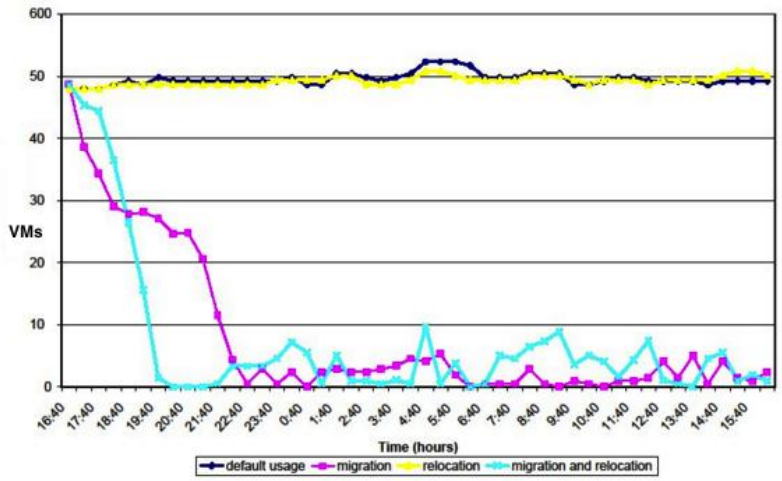

Figure 4. Load migration on cloud

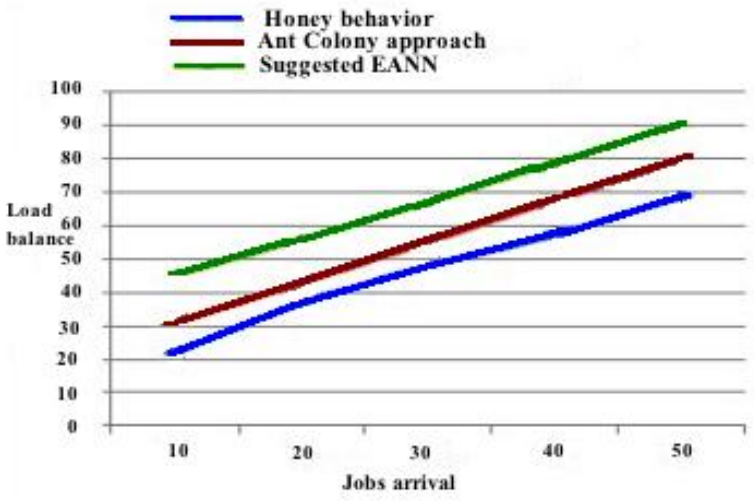

Figure 5. Load balancing comparison

4. CONCLUSION

Load prediction is needed for allocating data center resources depending on applications demands. The fluctuation of demand will be less detected using our approach. The system is managing resources and avoids overload. The suggested technique uses some of the design processes of a load balancing. EANN forecast the request then accordingly assign resources according to the request. That maintains the active servers according to current demand. This paper, suggested a new effective load balancing technique-using EANN based load-balancing technique to achieve detectable enhancements in resource deployment and availability among cloud-computing atmosphere.

For future, to further improve the balancing performance of load in cloud environments, for instance, the proactive job allocation method could be extended to consider multiple running tasks for simultaneous preemption and would increase the probability for high-priority tasks to start execution earlier and thus improve the overall system performance. For the shortest- workflow-first policy, other metrics for prioritizing workflows could be investigated in addition to the remaining execution time used in this paper.

REFERENCES

[1] R. P. Mahowald, Worldwide Software As A Service 2010–2014 Forecast: Software Will Never Be Same, In, IDC, 2010.

[2] Nuñez D, Fernandez – Gago C, Pearson S, Felici M. "A Metamodel for Measuring Accountability Attributes in the Cloud," In: *Proceedings of the 2013 IEEE International Conference on Cloud Computing Technology and Science (CloudCom 2013), IEEE Proceedings of the third International Conference on Cloud Computing*. IEEE, Miami, pp 297-304, 2013.

[3] Chazalet A., "Service Level Checking in the Cloud Computing Context," In: Wieder P, Butler JM, Theilmann W, Yahyapour R (eds) (2011) *Service Level Agreements for Cloud Computing*. Springer, 2010.

[4] C. Lim, S. Lu, A., Chebotko, and F. Fotouhi, "Storing, Reasoning, and Querying Opmcompliant Scientific Load Provenance Using Relational Databases," *Future Generation Computer Systems* 27, pp. 781–789, 2011.

[5] H. Topcuoglu, S. Hariri and M. Wu, "Performance-effective and Low-Complexity Task Scheduling for Heterogeneous Computing," *IEEE Trans. on Parallel and Distributed Systems*, vol. 13, pp. 260-274, 2002.

[6] J. D. Ullman, "NP-complete Scheduling Problems," *Journal of Computer and System Sciences*, Vol. 10, Iss. 3, pp. 384–393, 1975. [4] E. K. Byun, Y. S. Kee, J. S. Kim, and S. Maeng, "Cost optimized provisioning of elastic resources for application loads", *Future Generation Computer Systems* 27, pp. 1011- 1026, 2011.

[7] T. N'takpe' and F. Suter, "A Comparison of Scheduling Approaches for Mixed-Parallel Applications on Heterogeneous Platforms," *Proc. the 6th International Symposium on Parallel and Distributed Computing (IS-PDC)*, Hagenberg, Austria, July 2007.

[8] Message Passing Interface, http://www.mpi-forum.org/, (2015.3)

[9] M. Barreto, R. Avila and P. Navaux, "The Multicluster Model to the Integrated Use of Multiple Workstation Clusters," *3rd Workshop on Personal Computerbased Networks of Workstations*, pp. 71–80, 2000.

[10] K. C. Huang, "On Effects of Resource Fragmentation on Job Scheduling Performance in Computing Grids", *2009 10th International Symposium on Pervasive Systems, Algorithms, and Networks*, pp.701-705, 2009.

[11] C. C. Hsu, K.C. Huang and F.J. Wang, "Online Scheduling of Load Applications in Grid Environments," *Future Generation Computer Systems* 27, pp. 860–870, 2011.

[12] Santanu Dam, Gopa Mandal, Kousik Dasgupta and Paramartha Dutta, "Genetic Algorithm and Gravitational Emulation Based Hybrid Load Balancing Strategy in Cloud Computing", *in proc. third International Conference on Computer, Communication, Control and Information Technology (C3IT), IEEE*, pp. 1-7, February 2015.

[13] ASKALON, http://www.dps.uibk.ac.at/projects/teuta/ (2015.3)

[14] DAGman, http://research.cs.wisc.edu/htcondor/dagman/dagman.html (2015.3)

[15] M. Wieczorek, M. Siddiqui, A. Villazón, R. Prodan, T. Fahringer, "Applying Advance Reservation to Increase Predictability of Workflow Execution on the Grid", Proceedings of 2nd IEEE International Conference on e-Science and Grid Computing, pp. 82, December 4-6, Amsterdam, Netherlands, 2006.

[16] Zeba Khan, Mahfooz Alam, Raza Abbas Haidri "Effective Load Balance Scheduling Schemes for Heterogeneous Distributed System," *International Journal of Electrical and Computer Engineering (IJECE)*, Vol.7, No.5, October 2017, pp. 2757~2765, ISSN: 2088-8708, 2017.

[17] R. Prodan, T. Fahringer, "Dynamic Scheduling of Scientific Workflow Applications on the Grid: A Case Study", *Proceedings of the20th Symposium on Applied Computing (SAC 2005)*, pp. 687-694, 2005.

[18] Kefaya Qaddoum, Evor Hines, and Daciana Iliescu, "Yield Prediction Technique Using Hybrid Adaptive Neural Genetic Network," *International Journal of Computational Intelligence and Applications*, Vol. 11, No. 04, 1250021, 2012, DOI: http://dx.doi.org/10.1142/S1469026812500216.

[19] T. Sasidhar, V. Havisha, S. Koushik, M. Deep, VK. Reddy, "Load Balancing Techniques for Efficient Traffic Management in Cloud Environment," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 6, no. 3, pp. 963-973, 2016.

[20] P. Qiao, "On the Security of a Dynamic and Secure Key Management Model Forhierarchical Heterogeneous Sensor Networks," *TELKOMNIKA Indonesian Journal of Elec-trical Engineering*, vol. 12, no.10, pp. 7459–7462, 2014. http://iaesjournal.com/online/index.php/TELKOMNIKA/article/view/5579

[21] Cao, X., Zhong, Y., Zhou, Y., Wang, J., Zhu, C., & Zhang, W. "Interactive Temporal Recurrent Convolution Network for Traffic Prediction in Data Centers," *IEEE Access*, 6, 5276-5289, 2018.

[22] Peng, G., Wang, H., Zhang, H., & Dong, J. "Knowledge-Based Resource Allocation for Collaborative Simulation Development in a Multi-Tenant Cloud Computing Environment," *IEEE Transactions on Services Computing*, 2016. DOI: 10.1109/TSC.2016.2518161

BIOGRAPHIES OF AUTHORS

Dr. Kefaya Qaddoum is an Assistant professor at Higher Colleges of Technology UAE, was an Assistant professor in Al-Ahliyya Amman University. She received her first degree in Computer Science from Philadelphia University, Jordan, in July 2003, master degree in Computer Information System from Philadelphia University July 2008, and a doctorate degree in A.I from Warwick University, the UK in 2013. Her research area of interest includes A.I, Machine Learning, Data Mining, and Networking.

Dr. Mosleh M. Abu-Alhaj is a senior lecturer in Al-Ahliyya Amman University. He received his first degree in Computer Science from Philadelphia University, Jordan, in July 2004, master degree in Computer Information System from the Arab Academy for Banking and Financial Sciences, Jordan in July 2007, and a doctorate in Multimedia Networks Protocols from Universiti Sains Malaysia in 2011. His research area of interest includes VoIP, Multimedia Networking, and Congestion Control. Apart from research, Dr. Mosleh M. Abu-Alhaj also does consultancy services in the above research areas and directs the Cisco academy team at Al-Ahliyya Amman University.

Nameer N. EL-Emam: He completed his Ph.D. in Computer Science with honor in 1997. He works as an assistant professor in the Computer Science Department, Basra University. In 1998, he joined the Department of Computer Science, Philadelphia University, as an Assistance Professor, and then he got the rank "Associate Professor" in 2010. Now he is a Full Professor at the same university. He works as a chair of Computer Science Department and the deputy dean of the faculty of Information Technology, Philadelphia University. His research interest includes Computer Simulation with the intelligent system, Parallel Algorithms, and Soft computing using Convolution Neural Network, GA, ACO, and PSO for many kinds of applications like Image Processing, Sound Processing, Fluid Flow, and Computer Security (Steganography).