I.J. Information Technology and Computer Science, **2015, 02, 28-34**
Published Online January 2015 in MECS (http://www.mecs-press.org/)
DOI: 10.5815/ijitcs.2015.02.04

A Swarm Intelligence Based Model for Mobile Cloud Computing

Ahmed S. Salama
Computer and Information Systems Department, Sadat Academy for Management Sciences, Cairo, Egypt
Email: ahmedsayedsalama@yahoo.com

Abstract—Mobile Computing (MC) provides multi services and a lot of advantages for millions of users across the world over the internet. Millions of business customers have leveraged cloud computing services through mobile devices to get what is called Mobile Cloud Computing (MCC). MCC aims at using cloud computing techniques for storage and processing of data on mobile devices, thereby reducing their limitations. This paper proposes architecture for a Swarm Intelligence Based Mobile Cloud Computing Model (SIBMCCM). A model that uses a proposed Parallel Particle Swarm Optimization (PPSO) algorithm to enhance the access time for the mobile cloud computing services which support different E Commerce models and to better secure the communication through the mobile cloud and the mobile commerce transactions.

Index Terms—Mobile Cloud Computing, Swarm Intelligence, Parallel Particle Swarm Optimization (PPSO), E-Commerce

I. Introduction

In the following three subsections, mobile cloud computing, swarm intelligence and literature review of these two concepts are introduced.

A. Mobile Cloud Computing

“Mobile Cloud Computing at its simplest refers to an infrastructure where both the data storage and the data processing happen outside of the mobile device. Mobile cloud applications move the computing power and data storage away from mobile phones and into the cloud, bringing applications and mobile computing to not just smartphone users but a much broader range of mobile subscribers”. MCC can be defined as a combination of mobile web and cloud computing [1], which is the most popular tool for mobile users to access applications and services on the Internet [2]. Aepona describes MCC as a new paradigm for mobile applications whereby the data processing and storage are moved from the mobile device to powerful and centralized computing platforms located in clouds. These centralized applications are then accessed over the wireless connection based on a thin native client or web browser on the mobile devices [3]. The term “mobile cloud computing” was introduced not long after the concept of “cloud computing” launched in mid-2007. It has been attracting the attentions of entrepreneurs as a profitable business option that reduces the development and running cost of mobile applications, of mobile users as a new technology to achieve rich experience of a variety of mobile services at low cost, and of researchers as a promising solution for green IT [4].

Mobile cloud applications move the computing power and data storage away from mobile phones and into the cloud, bringing apps and mobile computing to not just smartphone users but a much broader range of mobile subscribers. In this section, we enlist the possible benefits of Mobile Cloud Computing [5].

- Mobile Cloud Computing will help to overcome limitations of mobile devices in particular of the processing power and data storage.
- It also might help to extend the battery life by moving the execution of commutation-intensive application ‘to the cloud’.
- Mobile Cloud Computing is also seen as a potential solution for the fragmented market of mobile operating systems with currently eight major operating systems.
- Mobile Cloud Computing can increase security level for mobile devices achieved by a centralized monitoring and maintenance of software,
- It can also become a one-stop shopping option for users of mobile devices since Mobile Cloud Operators can simultaneously act as virtual network operators, provide e-payment services, and provide software, data storage, etc. as a service.
- A number of new technical functionalities might be provided by mobile clouds. In particular, provisioning of context- and location-awareness enables personalization of services is an attractive functionality.
- Mobile Cloud Computing might open the cloud computing business that is currently almost exclusively addressing businesses to consumers since they will significantly benefit from the above described options.

B. Influence of Mobile Cloud Computing Upon E-Commerce

The Influence of the Cloud Computing upon the E-Commerce can be summarized as follows: [6],[7]

- Cloud Computing Can Provide Good Economic Efficiency for the E-Commerce Application. Making good use of the cloud technology application in the business field, and taking effective use of resources will lead to reduced costs. Thus it will enchant enterprise's kernel competitive power and eventually finished goods and services trading.
- It is not only can provide service for multiple clients at the same time, but also can guarantee a high degree of security of its application environment.

- It is no longer to pay for the high cost of software and application development, only need to assign the task to any free IT resources in the clouds and to help it fulfill the task.
- Cloud Computing Can Improve the Vivid and Profession of Business Enterprise E-Commerce Application. use the cloud computing platform, virtual constructing the distributed resources in all different places, realize the resource sharing of application, there is no need for the enterprises to know the details of sharing and how to achieve it.
- With the advent of cloud computing, existing businesses and startups can immediately leverage the trust built into established cloud systems such as Google, Amazon and Salesforce.
- Cost is generally one of the primary reasons for moving a business application or data center to the cloud. A cloud-based initiative on a virtualized server may save a company 80% of the costs normally associated with a traditional e-commerce roll out.
- A company may be able to roll out an e-commerce application five times faster than before and begin selling immediately on the remote platform.
- Cloud services allow a business to scale quickly and support seasonal spikes in demand or those triggered by special promotions.
- Cloud computing offers many security technologies for the running applications on the cloud. Cloud Computing Can Improve the Safety of Business Enterprise E-Commerce Application
- The explosive growth in cloud ecommerce offerings in the next few years will also see an increase in the ability to share information between clouds and communities of clouds.

C. Swarm Intelligence and Cloud Computing

Swarm intelligence (SI) is the combined behavior of decentralized, self-organized systems, natural or artificial. The concept is engaged in effort on artificial intelligence. The expression was originally initiated by “Gerardo Beni” and “Jing Wang” in 1989, in the perspective of cellular robotic systems. Swarm assumption has been built-up and discussed in. Swarm behavior involves dynamism. SI systems are typically made up of a population of ordinary agents cooperating locally with one another and with their environment. The inspiration often comes from nature, especially biological systems [8].

The agents follow very simple rules, and although there is no centralized control configuration dictating how individual agents should act, local, and to a certain degree arbitrary, communications among such agents direct to emergence of “intelligent” global behavior, unknown to the individual agents. Natural examples of SI include ant colonies, bird flocking, animal herding, bacterial growth, and fish schooling [9].

Group occurs at all magnitudes from microorganisms to whales, from factions of 10 to 10 million, and across a variety of sequential permanence from the transient congregation of midges to the mandatory regulations of herring. An aggregation may outline initially by arbitrary encounter and produce by density based connections. Group size is then determined by the balance of payoffs to individual members, where size of the group affects its performance [10].

In heterogeneous system environment, several types of servers can be implemented for better selection of tasks to be compiled and executed for heterogeneous supercomputing. High performance can be achieved using cluster of workstations [11]. These workstations can be of similar types or distinct types in respect of configurations. Therefore, practical situation of the distributed network needs multi-processing power within the network activities to speed-up the task with synchronization [12]. Whenever a transaction is taking place in a server-side machine for a particular process, all other processes would wait till its completion [13]. Another important point is transactional memory and garbage collection comparison which is dealt in [14]. In distributed systems, shared memory concept should be applied for multi-processor operating systems. In this way, the network memory can be considered as local memory [15]. Proper management of central processing unit, memory and network bandwidth is required while the external users access server-side resources having limitations [16]. Concurrent execution of procedures has been accomplished in the network scenario maintaining specific data structures [17].

This paper is organized as follows: section 2 introduces a proposed architecture of swarm intelligence based model for mobile cloud computing, section 3 presents Proposed PSO & PPSO Based Algorithms for Mobile Cloud Computing. Section 4 summarizes the required implementation issues for the proposed architecture and algorithms, section 5 presents the conclusions, and finally, section 6 gives a look forward for the expected future work in this domain.

II. Proposed Swarm Intelligence Based Model for Mobile Cloud Computing Architecture

This paper introduces architecture of swarm intelligence based model for mobile cloud computing and distributed computing environment. It can be considered as a base for running mobile ecommerce applications over mobile clouds. The proposed architecture is shown in figure 1.

The mobile device (cloud client) communication services are responsible for executing different network communication management between the mobile cloud client and the cloud. These routine services include:

- Synchronizing state changes made to the mobile or its applications back with the Clouds Servers
- Managing any state updates being sent as a notifications from the cloud server
- Managing the communication channel needed to receive any notifications coming from the clouds servers and establishing proper connections automatically.

- Applying the chosen security protocol and any required encryption and decryption processes to the sent and received data going out and received by the mobile cloud client.

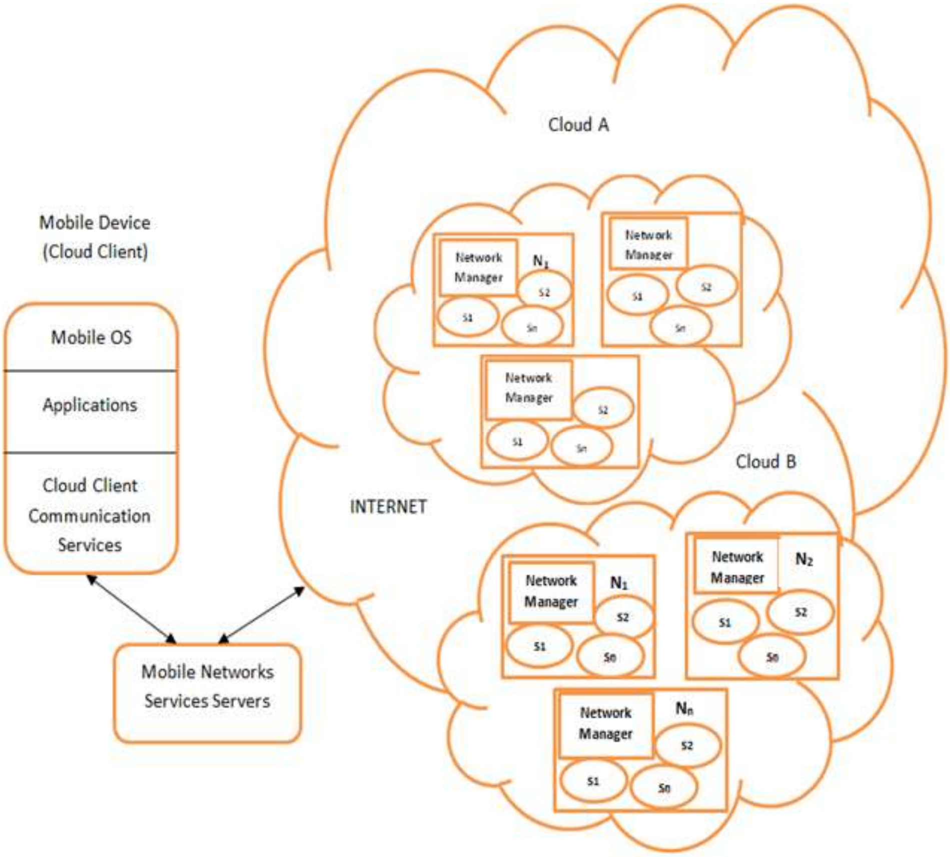

Fig. 1. Swarm Intelligence Based Model for Mobile Cloud Computing Architecture

Mobile users' requests and information (e.g., ID and location) are transmitted to the servers providing mobile network services. Mobile network operators can provide the authentication, authorization and accounting services (AAA). After doing these functions, the requests are delivered to clouds through the Internet. Each cloud consists of a set of networks N = {N_1,N_2,….,N_n}. Each network may have several application servers (SAs) S = {s_1,s_2, ……., s_n} providing the Software As A Service (SAS) and can be allocated to execute the mobile cloud clients' requests. Each application server has a set of resources R = {r_1,r_2,………,r_n} that can be allocated for the coming mobile clients' requests. Each Network N_i has a network manager that is responsible of the supervising and coordination of the communication between the servers inside the network and between this network and other networks in the clouds. Each Server s_i inside network N_i detects the geographical location of mobile cloud client MCC_i using user IP.

III. Proposed PSO & PPSO Based Algorithms for Mobile Cloud Computing

In the proposed architecture, a proposed PSO (Particle Swarm Optimization) based algorithm for mobile cloud computing is introduced. In addition, for enhancing the efficiency of the cloud computing, a proposed PPSO (Parallel PSO) algorithm for mobile cloud computing is also introduced and the recommended communication strategy is determined for implementing this PPSO algorithm.

A. Updated PSO Algorithm for Cloud Computing

An updated PSO Algorithm for Cloud Computing is proposed as follows (A declarative flow chart is shown in Figure 2):

Assume that there are M particles (servers) in all the clouds. Each server si in the cloud(s) is considered a particle which represents a potential solution (server) that can be allocated for executing the client's subjobs ji. Let J = {j1,j2,…..,jn} be the n user's subjobs to be scheduled in unit time. The meanings of the used symbols are:

$v_i(t)$: the ith particle velocity in iteration t;

$x_i(t)$: the particle i position;

c1, c2: constant weight factors;

$b_i(t)$ (lbest): particle with the local best calculated fitness value among the neighbored particles in the network achieved so far in the t-th iteration ;

G(t) (gbest): particle with the global best position found in the cloud or all the clouds;

r_1, r_2: random factors in the [0, 1] interval;

Input: The user's job vector J.

Output: Results of executing the user' job returned to the ISP server to send it back to the user's client device (c_i).

So the PSO on mobile cloud computing can be applied as follows:

a) Initialize the population with M Particles (servers S) and Set iterations counter t = 0.

b) **While not all** user's subjobs j_i **in** J **Not Completed**

c) Apply Fitness function: Calculating the fitness value by calculating the percentage of this particle (server) will share in minimizing the makespan and thus minimizing the total processing time for user’s job, and calculating the percentage of resources utilization if this server s_i is assigned the client’s job with the aim of maximizing each particle’s (server’s) s_i resources utilization.

d) Compare the calculated fitness value of each particle with its lbest $b_i(t)$. If current value is better than lbest, then set the current location as the lbest location. Furthermore, if current value is better than gbest G(t), then reset gbest to the current index in particle array. Select the best particle as gbest.

e) Update each Particle Velocity according to Eq. (1).

$$v_i(t+1) = v_i(t) + c_1.r_1.\left(b_i(t) - x_i(t)\right) + c_2.r_2.\left(G(t) - x_i(t)\right) \tag{1}$$

where i= 0,1,2,…,M-1

To prevent the velocity from becoming too large, we set a maximum value to limit the range velocity as

$$-V_{MAX} \leq V \leq V_{MAX}$$

f) Update each Particle position according to Eq. (2).

$$x_i(t+1) = x_i(t) + v_i(t) \tag{2}$$

where i= 0,1,2,…,M-1

g) Master server assigns user's subjob j_i to the server s_i with the highest fitness value in the t-th iteration and which has a current position (x_i).

h) t = t +1.

i) **End While.**

j) Stop.

So these recursive steps continue until we reach the termination condition (which is the user job is finished execution) and the results are ready to be sent back to the user’s client directly, or sent back for the master server to aggregate the results coming from all particles executing all the user’s jobs that form the running user’s application on the user’s client (smart cellular phone for example).

B. Proposed PPSO Algorithm for Mobile Cloud Computing

a) The master server (Network Manager) receives the user job (Application/Module with its status data (user IP, time stamp).

b) The master server (Network Manager) divides the received user's job (Task) J into subjobs {j_1,j_2,j_3, ….. , j_n}.

c) The user job is broadcasted on the web by the master server.

d) Each server has its own information about the available resources and about other servers within specific network. The master server (Network Manager) maintains the information about all other related networks within the internet providing SaaS.

e) Apply the PSO algorithm.

f) The results of the user job (j) will be sent back to the ISP server to send it back to the user’s client device (c_i).

g) Network N_j controller transmits the update to other related networks.

h) Each related network updates the related servers.

i) Stop.

In PPSO, computation time of PSO can be reduced with the parallel structure. Parallel processing aims at producing the same results achievable using multiple processors with the goal of reducing the run time. The same steps described in PSO will be applied, but in step

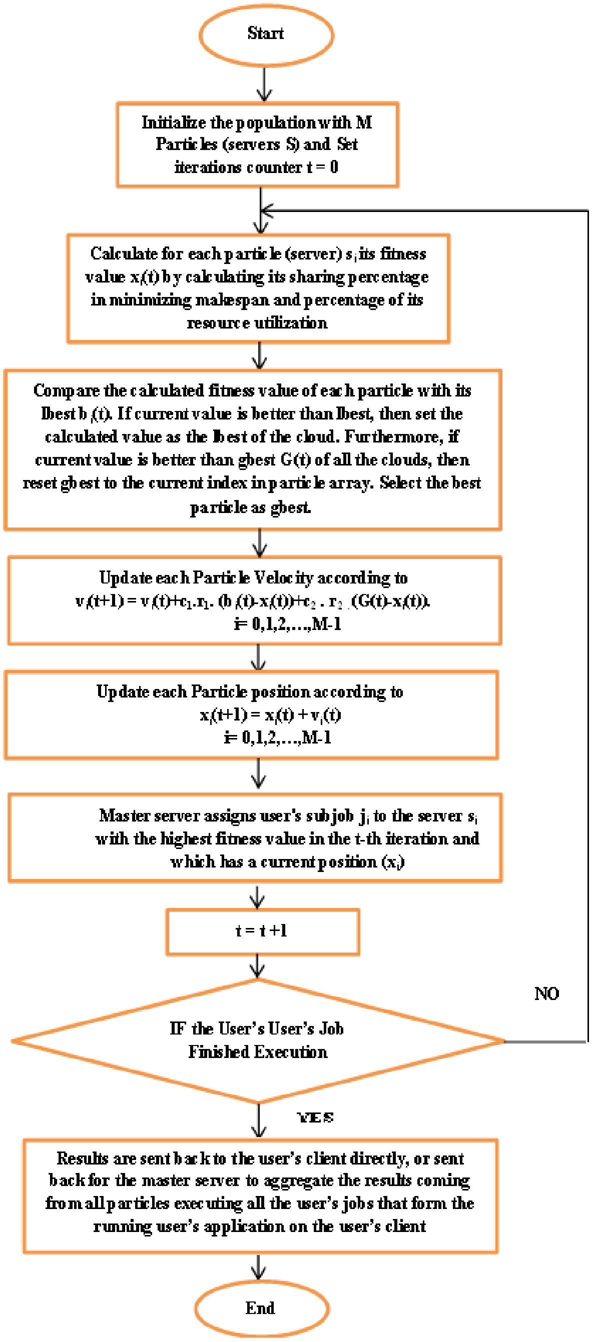
Start
Initialize the population with M Particles (servers S) and Set iterations counter t = 0
Calculate for each particle (server) s_i its fitness value $x_i(t)$ by calculating its sharing percentage in minimizing makespan and percentage of its resource utilization
Compare the calculated fitness value of each particle with its lbest $b_i(t)$. If current value is better than lbest, then set the calculated value as the lbest of the cloud. Furthermore, if current value is better than gbest G(t) of all the clouds, then reset gbest to the current index in particle array. Select the best particle as gbest.
Update each Particle Velocity according to $v_i(t+1) = v_i(t)+c_1.r_1.(b_i(t)-x_i(t))+c_2.r_2.(G(t)-x_i(t))$. i= 0,1,2,…,M-1
Update each Particle position according to $x_i(t+1) = x_i(t) + v_i(t)$ i= 0,1,2,…,M-1
Master server assigns user's subjob j_i to the server s_i with the highest fitness value in the t-th iteration and which has a current position (x_i)
t = t +1
IF the User’s User’s Job Finished Execution
NO
YES
Results are sent back to the user’s client directly, or sent back for the master server to aggregate the results coming from all particles executing all the user’s jobs that form the running user’s application on the user’s client
End

Fig. 2. Proposed PSO Based Algorithm for Mobile Cloud Computing

(a), the master server in specific network will define how many group of servers needed for the user's application to be executed, because it can be designed to be 2n sets. To decide the number of groups, it should be carefully refer to the size of the user's application with its internal jobs (modules) to execute. In general, use four groups of servers in the cloud, but use eight groups in the cloud only if the user's application is quite large.

The performance of the Parallel PSO can be evaluated using Amdahl's Law Eq. (3). [18] Let p be the fraction of the user's subjob j_i that can be made parallel (p is always a fraction less than 1.0.) The remaining fraction (1-p) of the code must run serially. n is the number of allocated CPUs or processing cores for j_i and these allocated resources r are considered subset of the resources vector R in the clouds, so we can represent them as $r \subseteq R$.

$$\text{Speedup (n)} = 1/((p/n)+(1-p)) \quad (3)$$

Suppose that p=0.8, and the number of used processors for parallelization (n) is 4, then then according to Eq. (3) Speedup equals 2.5. Also, if the number of used processors for parallelization increased to 8 Speedup will equal 3.3. The maximum possible speedup—if you could apply an infinite number of CPUs—would be 1 / (1-p). There is a proportional relationship between increasing the value of fraction p and the possible increasing of speedup, as shown in the graph in Figure 3.

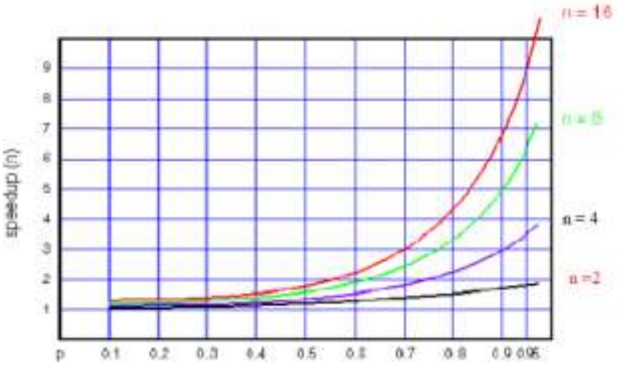

Fig. 3. Possible Speedup Values for Different Values of P and No. of Processors

After several iterations, each group of servers in the cloud will exchange the information about the fitness value of all the servers in the group and the status of completed user's jobs in the group. This gives every group of servers the chance to have the global best solution (best chosen candidate servers across all the groups of servers) and the local best solutions of other groups (the best candidate server in each group). This will enhance the chance of finding a better server to be allocated across all the groups and even a better local candidate server to allocate inside each group.

The chosen communication strategy for the applied PPSO on cloud computing is as follows:

After a period of time is elapsed (t_i iteration), one of the following communication strategy will be executed according to a trade-off between the best particle in the group and the best of all other groups

a) Exchange the local best solution from one group to the others, OR
b) Apply a self-adjustment in each group of servers due to the strong correlations of parameters (the current fitness status of free available resources suitable for the user's job). In each group, the best candidate server will be migrated and allocated to the job replacing the other deteriorated server.

IV. Implementation

Interoperability becomes an important issue when mobile users need to interact and communicate with the cloud. HTML5, HTML5 WebSockets, java, and XML are used to implement the proposed PSO and PPSO algorithms for MCC and all the running communication services software proposed in the architecture on the mobile client devices and on the application servers to guarantee a high degree of interoperability between the mobile client devices and the clouds' servers. These implemented communication services also enabled data caching through a mobile device, and this allowed a cloud application to continue working if connection has been momentarily lost.

CloudSim as an experimental environment for cloud computing was used to simulate the proposed architecture for the swarm intelligence based mobile cloud computing model. It was used with an implemented algorithm for the proposed PSO algorithm in this paper to examine the performance of the proposed PSO and PPSO based algorithms for mobile cloud computing. Many experiments and tests were done for the setting of the PSO algorithm related to the independent parameters including the population size (no. of servers), no of iterations, c1, c2, c3, velocity weight at the start, and velocity weight at the end, etc. Table 1 summarizes some of the parameters settings for the proposed PSO algorithm in one of its configuration environment.

Table 1. Parameters Settings for the PSO Algorithm

Parameter	Value
Particles No.	20
Max No of Iterations	1000
C1	2
C2	2
C3	1
W_start	0.95
W_end	0.4
Chi	1
No. of Resources (processors)	32
No of Subjobs	32

When the number of VMs and the parallelization percentage were increased with applying the heuristic of choosing the particles (servers) that had a high predicted percentage in sharing in minimizing the makespan, the makespan was decreased as shown in figure (4). In addition, the resource utilization percentage was increased, when we allocated the servers that had a high predicted percentage of resources utilization if they are allocated to the user's subjobs j_i.

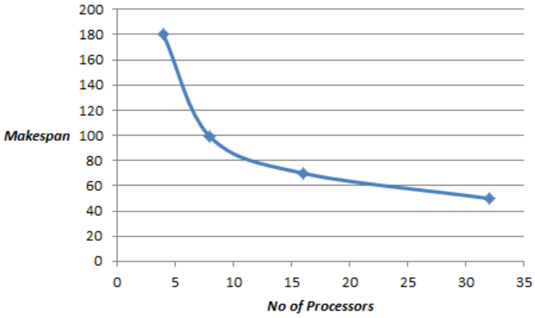

Fig. 4. Proposed PPSO leads to Reduced Mekespan with increasing in no. of processors

V. Conclusions

Mobile Cloud computing provides a capability of developing enhanced mobile applications, since it allows the transfer of the processing overhead to the clouds over the internet. In this paper, synchronized system networking concept has been offered using the proposed parallel particle swarm optimization (PPSO) algorithm across many different networks with several application servers. Architecture for the swarm intelligence based mobile cloud computing model was also proposed. The architecture is based on a proposed algorithm for the Particle Swarm Intelligence (PSO) and Parallel Particle Swarm Optimization (PPSO) to reduce the run time required by the mobile client's application on the different applications servers in the clouds and to enhance the applied security technologies ranges between authentication, authorization, and encryption. The proposed architecture and algorithms in this paper will improve the speed of running and the safety of Business Enterprise E-Commerce Applications. Many implementation tools such as HTML5, Java, XML, and HTML5 WebSockets are used for implementing the proposed intelligent swarm algorithms for the MCM, and all the communication services routines used to manage the interface between the mobile client devices and the application servers on the clouds. The proposed architecture and algorithms are expected to be intelligent, reliable, efficient, robust, and self-organizing in enhancing the SAAS that is provided on MCC.

VI. Future Work

Considering the importance of Mobile Cloud Computing, other intelligent communication techniques and more adequate security measures have to be incorporated to support the low processing ability at the mobile client-side, mobile commerce, and the distributed mobile computing environment models and architectures.

References

[1] H. Jacson Christensen, "Using RESTful web-services and cloud computing to create next generation mobile applications", Proceedings of the 24th ACM SIGPLAN conference companion on Object oriented programming systems languages and applications (OOPSLA), 2009, 627-634.

[2] L. Liu, R. Moulic, and D. Shea, "Cloud Service Portal for Mobile Device Management", Proceedings of IEEE 7th International Conference on e-Business Engineering (ICEBE), 2011, 474.

[3] AEPONA, "Mobile Cloud Computing Solution Brief", White Paper, (2010).

[4] M. Ali, "Green Cloud on the Horizon", Proceedings of the 1st International Conference on Cloud Computing (CloudCom), 2009, 451- 459.

[5] S. Chetan, K. Gautam, "Cloud Computing for Mobile World", Technical Report, 2010.

[6] C. Sun, "Research of E-Commerce Based on Cloud Computing", Advances in Computer Science and Information Engineering, Vol. 169, 2012, 15-20.

[7] S. Juncai, and Shao, "Based on Cloud Computing E-commerce Models and Its Security", International Journal of e-Education, e-Business, e-Management and e-Learning, Vol. 1, No. 2, 2011, 175-180.

[8] A. Kundu, and Ji. Chunlin, "Swarm Behavior of Intelligent Cloud", arXiv preprint arXiv:1203.1395, 2012.

[9] E. Ridge, D. Kudenko, D. Kazakov, E. Curry, "Moving Nature-Inspired Algorithms to Parallel, Asynchronous and Decentralised Environments," in Self-Organization and Autonomic Informatics (I), Vol. 135, 2005, pp. 35-49.

[10] K. Carolin, K. Johannes, B. Freimut, "Swarm Intelligence for Analyzing Opinions in Online Communities," Proceedings of the 43 rd Hawaii International Conference on System Sciences, 2010, pp. 1-9.

[11] S. Pakin, "The Design and Implementation of a Domain-Specific Language for Network Performance Testing," IEEE Transactions on Parallel and Distributed Systems, Vol. 18, No.10, 2007, pp. 1436-1449.

[12] E. Hart, Thomas, E. Paul McKenney, and D. Angela Brown, "Making lockless synchronization fast: Performance implications of memory reclamation," In 20th IEEE International Parallel and Distributed Processing Symposium , IPDPS 2006, 2006.

[13] M. Herlihy, "The transactional manifesto: software engineering and non-blocking synchronization," In PLDI '05: Proceedings of the 2005 ACM SIGPLAN conference on Programming language design and implementation, New York, NY, USA, 2005, pp. 280-280.

[14] D. Grossman, "The transactional memory / garbage collection analogy," In OOPSLA '07: Proceedings of the 22 nd annual ACM SIGPLAN conference on Object oriented programming systems and applications, New York, NY, USA, 2007, pp. 695-706.

[15] B. Gamsa, Orran Krieger, J. Appavoo, and M. Stumm, "Tornado: Maximizing locality and concurrency in a shared memory multiprocessor operating system", In Proceedings of the 3rd Symposium on Operating System Design and Implementation , New Orleans, LA, Vol. 99, 1999, pp. 87-100.

[16] B. Urgaonkar, P. Shenoy, "Sharc: Managing CPU and Network Bandwidth in Shared Clusters," IEEE Transactions on Parallel and Distributed Systems, Vol. 15, No. 1, 2004, pp. 2-17.

[17] M. Herlihy, "A methodology for implementing highly concurrent data structures," In Proceedings of the 2nd

ACM SIGPLAN Symposium on Principles and Practice of Parallel Programming , 1990, pp. 745-770.

[18] M. D. Hill, M. R. Marty, "Amdahl's Law in the Multicore Era," IEEE Computer Society, July 2008, pp. 33-38

Author’s Profile

Ahmed S. Salama received his BSc degree in computer sciences and information systems from Sadat Academy in 1991 Egypt, M.Sc. in Information Technology from Alexandria University in 1998, and his Ph.D. In information technology in 2004 from Alexandria University Egypt, He is mainly interested in machine learning applications, and knowledge discovery in Bid Data.

He worked as an assistant professor in information systems department in faculty of computer sciences – Taiba University – KSA, and now is working as an assistant professor in information systems department in Sadat Academy.

How to cite this paper: Ahmed S. Salama,"A Swarm Intelligence Based Model for Mobile Cloud Computing", International Journal of Information Technology and Computer Science(IJITCS), vol.7, no.2, pp.28-34, 2015. DOI: 10.5815/ijitcs.2015.02.04