Three Way Handshaking Method: New Theoretical Way for Improving Distributed Computing

Hamid Zargari Asl[1], Behzad Sami[2], Mostafa Dashti[3], Farzad Nourmohammadzade[4]
[1]Institute of Information Technology of ANAS, Baku, Azerbaijan
[1-4]Payam e Noor University, Ardebil , Iran
[1]*hamid_zargari@yahoo.com*, [2]*behzad.sami@yahoo.com*, [4]*farzad.motlag@yahoo.com*

***Abstract*— Distributed computing has a special role in computer world, especially when we want to do massive process. The special advantage of distributed computing is its speed and unfortunately sometimes this speed due to lack of agents or algorithmic mistakes decrease. In this paper we introduce a new method for preventing of speed decreasing by using three way handshaking, and results compared to each other before and after using this technique. Finally, some suggestions are presented.**

Keywords*— *distributed computing; speed improving; three way handshaking

I. INTRODUCTION

Computer software traditionally ran in standalone systems where the user interface, application 'business' processing and persistent data resided in one computer with peripherals attached to it with cables. Instead of it, most important computer software today runs in distributed systems. A distributed system is a computing system in which a number of components (mobile agents) cooperate by communication over a wireless network. The field of distributed computing originates from parallel computing and belongs to the study of concurrent systems. Figure 1 shows a typical architecture of distributed computation.

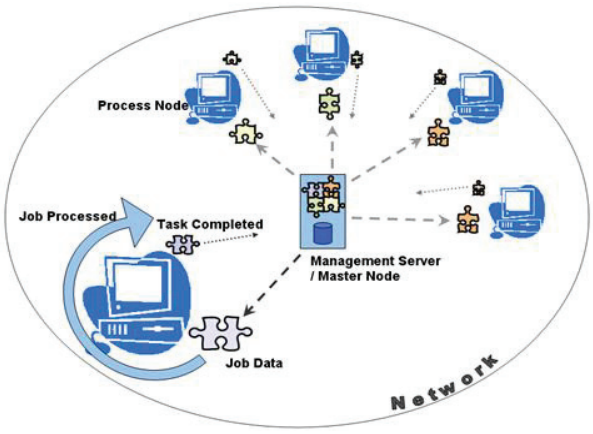

Figure 1. Distributed computing systems

The first wide spread distributed systems were local area network (LAN) such as Ethernet that was invented in the 1970s and the study of distributed computing became its own branch of computer science in the late 1970s and early 1980s. The classical distributed computing has some problems, such as network traffic and network latency. So, today, by introducing of software like Mobile Agent, are wanted to address these problems.

A mobile agent is an agent that's not bound to the system where it begins execute. It has unique ability to transport itself from one system in a network to another. The using of these agents by central server needs some algorithms and ability of knowing the idle agents. Unfortunately, most of distributed computing systems have problem with these issue and some solutions have been suggested for it.

II. RELATED WORKS

A lot of studies have been done in field of distributed computing and its implementation or the techniques about better using of mobile agents and in this way some works relate to our work. For instance, Rushikesh et al use a particular pattern in order to make a secure and fast way for distributed computing by mobile agent abstraction [1] [2]. At the other side, Douglas C.Schmidt believes that key characteristics of distributed systems are more efficient in their utilizations [3]. Some of the scientists declare that mobile agent network design is so important in distributed computing [4] [5] [6] [7] [8]. In all of this essays, placement of agents is clear, but what would be happened when their correct placement were unknown?

III. DISCUSSION

Current developments of network, especially mobile networks, make the usage of distributed computing simple. Today's large scale computational requirements aren't easy to be handled. In order to have a smooth and reliable communication, and also for maintaining our high speed computation, we are recommending three way handshaking method in distributed computing. In which, we have a central server that surrounded by mobile agents that we don't know about number and locations. Our idea is that, when we have a computation process which wants to be distributed and get back the result, it sends a request from central server to all mobile agents to do it. Based on from which part of network we have received more numbers of acknowledgments, we send the process over there and let them to do it and send

back consequence. After receiving the result, in turn, we send our acknowledgment to them that we got it.

According to increase of procedures volume in the computer systems that have been executed, it seems comfortable that the usage of distributed computing for improving the speed of functions have been done. On the other hand, lack of recognition of mobile agent's placement and lack of identification of ready systems for working reduces the speed of functions. Therefore, our method is recommended. Using this method, the speed of functions is increasing until the expected level is reached. As it can be seen, by increasing the number of process (N.O.P), system utilization decreases. But, using our method causes it to be strongly increased. They are completely illustrated in Figures 2 and 3.

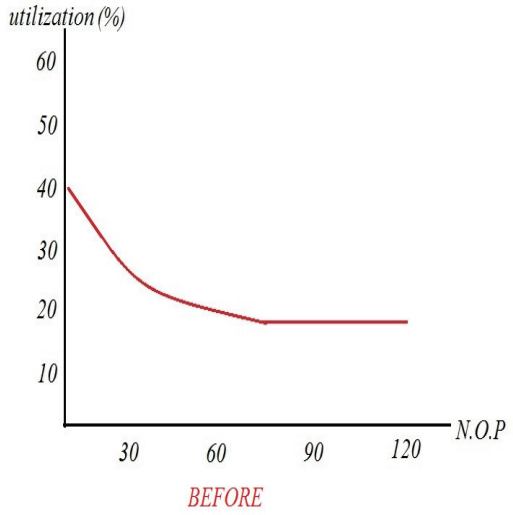

Figure 2. Distributed computing before our method

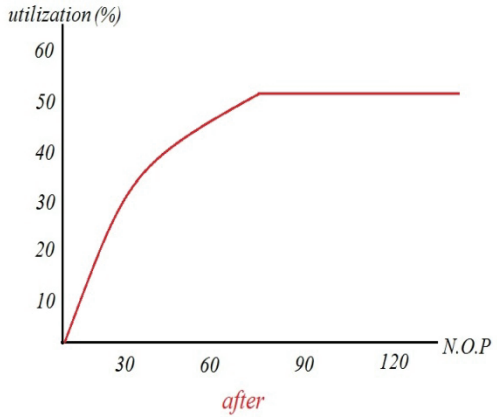

Figure 3. Distributed computing after our method

REFERENCES

[1] Rushikesh K Joshi, Harikrishnan C R, M. , Hidayat Ansari: Mobile Agent Abstractions: Formulations and Implementations, MTCOORD 2007, www.cse.iitb.ac.in/~rkj/mtcoord.pdf.

[2] Hosoon Ku, Gottfried W.R. Luderer and Baranitharan Subbiah: An Intelligent Mobile Agent Framework for Distributed Network Management, Network System Laboratory Telecommunication Research Center, Arizona State University, Global Telecommunications Conference, vol.1, GLOBECOM '97, 1997, pp 160 - 164.

[3] Douglas C.Schmidt: Software Technologies for Developing Distributed Systems: Objects and Beyond, Vanderbilt University, 2009, www.dre.vanderbilt.edu/~schmidt/PDF/CSI-article.pdf

[4] Danny B. Lange: Mobile Objects and Mobile Agents: The Future of Distributed Computing? , ECOOP, Brussels, Belgium, 1998, pp1-12.

[5] www.ifs.uni-linz.ac.at/~ecoop/cd/papers/1445/14450001.pdf

[6] Gregory Neven, Erik Van Hoeymissen, Bart De Decker, Frank Piessens: Implementing Secure Distributed Computing with Mobile Agents, Department of Computer Science, KU Leuven, Belgium , 2006.

[7] Eric Korpela, Dan Wertheimer, David Anderson, Jeff Cobb, Matt Lebofsky: SETI@HOME-- MASSIVELY DISRTIBUTED COMPUTING for SETI, Computing in Science & Engineering 2001, pp 78-83.

[8] Idit Keidar: Distributed computed column 36 Distributed Computing, 2009, ACM SIGACT, Volume 40, Issue 4, December 2009, pp. 64-67.

[9] Christian Ulrik Sottrup, Jakob Gregor Pedersen, Developing Distributed Computing Solutions Combining Grid Computing and Public Computing, M.Sc. Thesis, Department of Computer ScienceUniversity of Copenhagen, 2005.

[10] http://www.scribd.com/doc/62301636/Developing-Distributed-Computing-Solutions-Combining-Grid-Computing-and-Public-Computing.