An agent based TCP/IP for Wireless Networks

B. Sasikumar[1], and V. Vasudevan[2]
[1] *Professor, Department of CSE, RMD Engineering College, Kavaraipettai*
[2] *Professor, Department of IT, A.K. College of Engineering*
thilsasi@yahoo.com

Abstract

Recently, wireless devices are being used in ubiquitous ways and wireless networks are becoming increasingly important in the global communication architecture. Most of the wireless networks have fixed infrastructure and the need for mobile wireless networks are growing rapidly, which requires higher data rates. Due to high data rate in wireless networks the TCP performance has a broad and significant impact on data applications, and is essential towards a future massive deployment of service-providing agents and their widespread social acceptance. The uses of extensive local retransmission mechanism in wireless networks are adequate. In order to improve the performance of TCP/IP, a JADE (Java Agent Development Environment) agent is used to implement along with the Transmission Control Protocol (TCP). The personal agent communicates wirelessly, in order to find the most appropriate option to serve the user. Performance has been analyzed for the TCP/IP with and without JADE agent platform as technology choice and investigated its efficiency in a number of cases.

Keyword

FIPA, latency, performance, JADE agent, spammer, TCP, throughput, wireless networks.

1. Introduction

A wireless network consists of mobile or stationary nodes which can communicate with each other over the wireless links without the aid of any established infrastructure or centralized administration. Each node has the capability to communicate with another node in its vicinity, which forwards the data packets to the designated node. These types of networks are useful in any situation where temporary network connection is needed, in case of disaster relief or in battlefield. Examples of these networks are in ad-hoc wireless networks, local area networks, packet radio communication, sensor networks etc.

In this era, citizens keep demanding fast, reliable, cheap, friendly and flexible ways of accessing any kind of information. They want to get the information from anywhere, anytime, using the latest technologies developed in the Information and Communication Technologies field, such as mobile phones, portable PCs or PDAs (Personal Digital Assistants) with the wireless technology. The current trend seems to go in the direction of joining these tools into single pieces of hardware with multiple capabilities.

Recently intelligent software agents [28] have a collection of properties that make them very adequate to provide services to citizens [1]. Some of the features are following:

- **Autonomy:** they can perform their tasks without a direct and continuous guidance from the user.
- **Learning:** they can apply machine learning techniques to construct automatically a user profile and adapt their actions to the user's preferences.
- **Proactiveness:** a personal agent can anticipate the needs of the user and perform tasks that may be beneficial for him, without an explicit request from the user.
- **Social ability:** a personal agent can get in touch with other agents that provide information about any domain in which the user may be intererested request the information that the user needs and present it in a friendly and personalized way.

Ad-hoc wireless networks are multihop wireless networks which consist of finite number of radio equipped nodes that are autonomous. In this each node has a transmission radius and is capable to transmit message to all its neighbors that are within the transmission range [19]. In multihop wireless networks the nodes are mounted on vehicles or carried by human beings. These networks concentrated on improving the Medium Access Control (MAC) layer protocols and routing protocols [24].

In broadcasting the source node sends the message to all the neighbor nodes which is called flooding, where each node retransmits a packets after receiving it, thus generates many redundant retransmission [5]. Some of the challenges that have to be taken care in a ad-hoc network are given [29].

In this paper we concentrated our attention to the transport layer routing which uses the Transmission Control Protocol (TCP). While running TCP over such networks, the throughput of the connection is viewed to be poor because this protocol takes the lost or delayed acknowledgements as congestion [13].

The reason for poor performance in Transmission Control Protocol has been examined in [25] and that need more effective mitigation strategies. We proposed a new agent to improve the performance of the TCP.

The rest of the paper is organized as follows. Section 2 describes the related work. Section 3 presents the performance of TCP/IP. Implementation of JADE agent is given in section 4. Section 5 dealt with the message exchange performances. Finally conclusion is given in section 6.

2. Related Work

TCP/IP networks with wireless networks can be described in three different ways namely gateway-based approach, middleware-based approach and TCP/IP for wireless networks approach. The sensor networks have its own communication protocols, which is simple to use the gateway with a TCP/IP network. There is no universal architecture suitable for all types of different wireless sensor networks [9]. When the users want to visit other types of sensor networks, they still need to redesign a new gateway. Another approach is using middleware to connect wireless sensor networks to IP based wire/wireless networks [26]. It is difficult to convert most kinds of protocols between sensor networks and TCP/IP networks using the VIP Bridge. In the future, a sensor node usually has more resources and more powerful computing capability, which will bring many novel changes to sensor networks. The drawbacks are not being able to support the mobility and flexible naming of sensor nodes [8], [30]. A TCP/IP Core for reconfigurable logic presented in [7] is able to process TCP/IP stack packets fully in dedicated hardware; other works [4], [20] also follow the same idea.

3. TCP/IP

The need on applications enabled by the Internet, such as electronic mail and web access, increases the popularity in business applications. The Transmission Control Protocol/Internet Protocol (TCP/IP) protocol suite is the engine for the Internet and networks worldwide. Its simplicity and power has led to its becoming the single network protocol of choice in the world today. TCP/IP was designed to connect multiple networks together in a seamless way.

Transmission Control Protocol is a reliable connection-oriented transport layer protocol that allows a byte stream originating on one node to be delivered without error on any other node in the network. It has emerged as a standard in data communication and this transport protocol uses a window based flow and error control algorithm on top of the Internet Protocol.

TCP reacts to any packet losses by dropping its congestion window size before retransmitting lost packets, initiating congestion control or avoidance mechanisms and backing off its retransmission timer [13]. Networks with wireless and other lossy links suffer from significant non-congestion losses due to handoffs and bit errors. This degrades the end-to-end performance by invoking congestion control and congestion avoidance algorithms.

The transmitting node identifies the packet loss either by the arrival of duplicate acknowledgement or by the absence of the acknowledgement for the packet within a timeout interval. When packets are lost in networks, due to congestion, results in an unnecessary reduction of end-to-end throughput and its related performance [27].

Most of the data applications use the TCP/IP, optimizing TCP performance over the network would have a broad and significant impact on the user perceived data application performance [21]. Increased delay variations which results in delay can cause timeouts in TCP, where the source incorrectly assumes that a packet is lost while the packet is only delayed, which forces TCP to slow start that reduce the TCP performance [22].

In wireless network the node connectivity tends to change over time. This rate may depend upon the factors like the number of nodes, velocity of mobile nodes, transmission range and obstacles in the environment. Due to the change in node connectivity two effects may occur [16]. Nodes may need to redirect its route through B for an ongoing TCP connection because node A has moved out of range of node C, which is shown in figure 1.

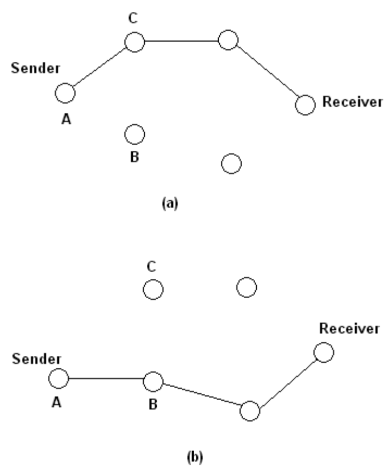

Figure 1. Route change due to mobility of node

Figure 2 shows that at different timings node A has an open TCP connection to node C. But the network gets portioned causing nodes A and node C to be in different portions. The change in node connectivity has tremendous consequences for TCP's throughput that will drop at very low levels. The impacts of these are described below [16], [18].

Bit Error Rate

These cause packets to get corrupted which results in lost TCP data segments or acknowledgements. For wireless networks the Bit Error Rate typically ranges from 12^{-2} to 10^{-6} [3].

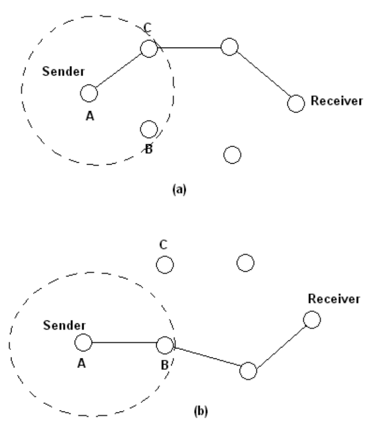

Figure 2. Partitions formed due to mobility of node

Route Recomputation

When an old route is no longer available, the network layer at the sender attempts to find a new route to the destination.

Network Partitions

If the sender and receiver of a TCP connection lie in different partitions, all the sender's packets get dropped by the network resulting in the sender invoking congestion control.

Multipath Routing

This minimizes the frequency of route recomputation, and results in significant number of out-of-sequence packets arriving at the receiver.

Latency

For high rate links leads to large bandwidth delay product which may disrupt the dynamic response to variations in link quality.

Window Translation Rate

In a TCP session through a bottleneck link, the congestion window is given by

$$W = R \cdot \Delta + \beta \qquad (1)$$

where,

R - fair share of the bottleneck rate
Δ - round trip propagation delay for the session
B - target buffer backlog for the session

If the fair rate for the session is time varying $R_{(t)}$, $\Delta_{(t)}$ an estimate at the transmitter of Δ at time t and $R_{(t-\Delta)}$ is the available rate as known to the transmitter at time t, then the naïve rate adapted window is

$$W_{(t)} = R_{(t-\Delta)} \cdot \Delta_{(t)} + \beta \qquad (2)$$

In this paper we attach a new agent named JADE to the TCP which improves the performance of TCP/IP. This agent reduces the problems at the congestion window and when the agent executes tasks, they organize themselves and may be smoothly running to avoid processing bottlenecks and form teams to deal with dynamic changes in information, tasks, number of agents and their capabilities.

4. JADE Agent

The key point towards a widespread acceptance of personal agents by citizens is its deployment in the wireless devices. If personal agents have to be able to communicate with other service providing agents, they must follow a set of standard norms concerning the agent communication language to be used, the communication protocols to be followed, etc.

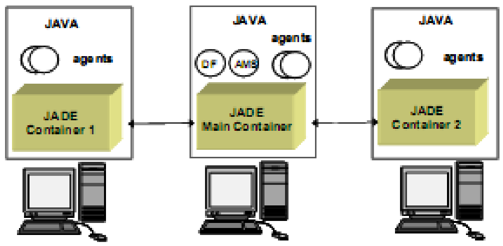

Figure 3. JADE main architectural elements

Java Agent Development Framework (JADE) [15], [10], is a software development framework aimed at developing multi-agent systems and applications conforming to Foundation for Physical Intelligent Agents (FIPA) [12], standards for intelligent agents that compliant multi-agent systems in wireless devices and is essential for the social acceptance of technology. It includes two main products: a FIPA compliant agent platform and a package to develop Java agents. JADE has been fully coded in Java and an agent programmer, in order to exploit the framework, should code the agents in Java. Figure 3 shows the main architectural elements of a JADE platform [23].

Each JADE system has a connection endpoint. A system can handle messages through any other connection. Connection binding provides an override of the default behavior so that the single user can handle all activity for a specific connection. The TCP/IP listen connection is automatically created and opened when the first user of the transport group invokes the begin listening method.

A user joins a transport group by creating a jade multiuser tcp transport instance, and when the last user leaves the group all connections are closed and the transport group is deleted. For multicast transport group it uses a separate instance of jade multicast user tcp transport instance for each group that joins. The client connection is idle, queued or assigned. To establish a new transport group or join an existing transport group, following code should be executed.

```
begin
   create self.myMWTT transient;
   self.myMWTT.listenPortnumber :=332
   21;
   self.myMWTT.beginListening();
   self.myMWTT.notifyEventsAsync(self
   ,JadeMultiWorkerTcpTrans
   port.Notify_Continuous);
end;
```

A connection is idle when it has no queued events such as data ready for reading. A connection is queued when it has one or more events ready for a worker to process. A connection is assigned when the user handles an event for the connection. Various exceptions can be raised when transport or connection properties are updated or methods are called.

All unhandled exceptions that cut back through a connection event method cause the connection to be closed. A TCP/IP connection treats the data being sent or received as a continuous stream of bytes. The application sending or receiving the data is responsible for deciding where one message stops and the next one begins.

5. Message exchange performance

In agent based system functionality is divided into agents [17], and these agents coordinate their actions and communicate by exchanging messages. If large amount of agents are used, large number of messages are to be expected. Here the message load efficiency of JADE agent is find out.

The general scheme of interaction between three spammer agents and three user agents is shown in the figure 4. This is designated to flood the system with Agent Communication Language messages [2]. In this spammer agents send to user agents a large number of messages.

All spammer agents start sending messages to all user agents. In the JADE platform all posted messages are put in a receiver message queue [14] and then it is processed by the receiver. During the execution each spammer agent broadcasts a certain number of messages and the total time of this broadcast is measured. A total of 50000 messages were sent by each spamming agent to each user agent.

Also we used the discrete simulator using wireless environment consisting of 50 wireless nodes roaming over a simulation area of 1500 meters x 1500 meters for 600 seconds of simulated time that propagates over a range of 250m and channel capacity was 2Mb/s at different pause times.

Each simulation scenario uses a total of 16 TCP connections, each are established for one of the configurations source sink pairs. Every source generates an infinite stream of data bytes. The parameters used in the simulation include the TCP packet size, the TCP window size and the switch buffer size.

Figure 5 shows the analysis of message sending and receiving times. This graph is plotted for the number of agent pairs along the x-axis and the time in milliseconds along the y-axis. From the analysis it is clear that the Spamming time is uniform even though the number of agent pairs is increased and in the case to receiving time it is linearly increasing for the increase in the number of agent pairs.

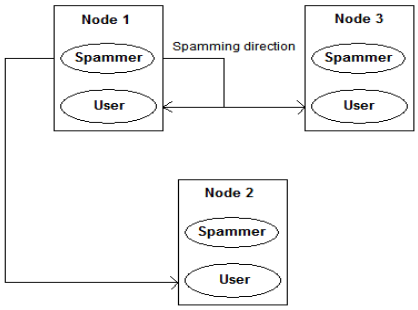

Figure 4. Spamming scheme

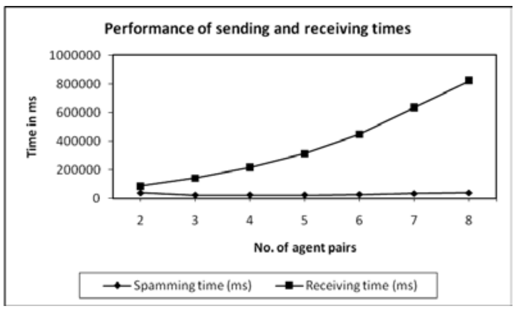

Figure 5. Analysis of message sending and receiving times

Figure 6 shows the performance analysis of throughput with respect to speed for the TCP/IP, with and without JADE agent. Analysis shows that the performance of TCP/IP is slightly good on mobility with JADE agent. In both the cases, as the mobility increase there is a decline in the throughput.

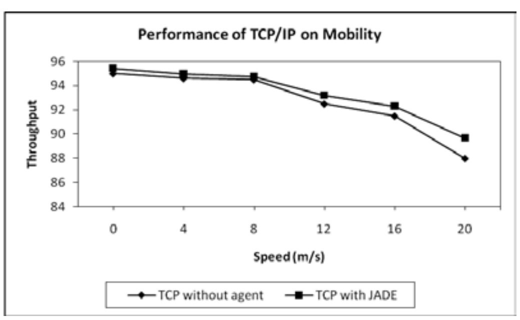

Figure 6. Analysis of Throughput

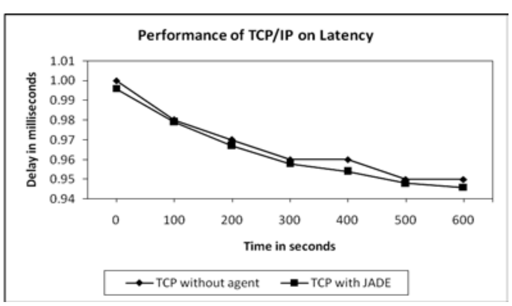

Figure 7. Analysis of Latency

Figure 7 shows the performance analysis of latency for the TCP/IP with and without JADE agent. Latency or response time is the time between the transmission of data packets from the source node and the time of its reception by a receiver node. From the analysis it is clear that initially the delay will be more and it get reduced gradually for both the case. Moreover the delay of TCP with JADE agent will be less than the delay of TCP without JADE agent.

6. Conclusion

The need for maintaining reliable and scalable platforms over network applications like web sites increases the proposed JADE agent based TCP which improved the performance in the wireless networks. By deploying JADE the crucial need for agents that make the administration and deployment in wireless networks becomes much easier. It also provides efficient reactive management systems. JADE is an excellent agent to depute development facility that provides full administration process with monitoring and reconfiguration. The analysis indicate that JADE is quite an efficient environment limited mostly by the standard limitations of Java programming language, which is interpreted and executed in a virtual machine, with processor speed, memory and network connection. An increase in the number or agents results typically in a linear increase of processing time.

7. References

[1] Agent Cities: http://www.agent cities.org.

[2] Agent Communication Language Specification, http://www.fipa.org/ respository/aclspecs.html.

[3] A. Bakre, and B. R. Badrinath, "I-TCP : Indirect TCP for Mobile hosts", in proceedings of 15th International Conference Distributed Computing Systems, Vancouver, BC, Canada, pp. 136-143, June 1995.

[4] Z. Bokai, Y. Chengye, "TCP/IP Offload Engine (TOE) for an SOC System", Institute of Computer & Communication Engineering, National Cheng Kung University, 2005.

[5] P. Bose, P. Morin, I. Stojmenovic, and J. Urrutia, "Routing with Guaranteed Delivery in ad-hoc wireless Networks", proceedings of third International Workshop Discrete Algorithms and methods for Mobile Computing and Communication (DIALM) pp. 48-55, Aug. 1999.

[6] G. Caire, LEAP 3.0 User Guide, TILAB.

[7] A. Dollas, I. Ermis, I. Koidis, I. Zisis, C. Kachris, "An Open TCP/IP Core for Reconfigurable Logic", Proceedings

of the 13th Annual IEEE Symposium on Field Programmable Custom Computing Machines (FCCM' 05), pp. 297- 298, 2005.

[8] A. Dunkels, T. Voigt, J. Alonso, “Making TCP/IP viable for wireless sensor networks”, Proceedings of the European Workshop on Wireless Sensor Networks (EWSN'04), Germany, 2004.

[9] A. Dunkels, T. Voigt, J. Alonso, H. Ritter, and J. Schiller: Connecting Wireless Sensornets with TCP/IP Networks, in Proceeding of the 2nd International Conference on Wired/Wireless Internet Communications, February, 2004.

[10] Edward Chen, Dorian Sabaz and William A. Gruver, “Wireless Distributed Systems with JADE”, IEEE International Conference on Systems, Man and Cybernetics, pp. 989-993, 2004.

[11] S. Floyd, “TCP and Explicit Congestion Notification”, http://www.nrg.ee.lbl.gov/floyd/ecn.html.

[12] Foundation for Intelligent Physical Agents: http://www.fipa.org.

[13] W. D. Ivancic, D. Brooks, B. Frantz, D. Hoder, D. Shell, D. Beering, “NASA’s Broadband Satellite Networking Research, IEEE Communications Magazine, pp.40-47, July 1999.

[14] JADE:http://Sharon.cselt.it/project s/ jade.

[15] JADE–LEAP:http://sharon.cslet.it/ project/ jade.

[16] Jain Liu, and Suresh Singh, “ATCP : TCP for Mobile Ad-hoc Networks”, IEEE journal on selected areas in Communications, Vol. 19, No. 7, pp 1300-1315, July 2001.

[17] N. R. Jennings, An agent-based approach for building complex software systems, Communications of ACM, 44 (4), pp. 35-41, 2001.

[18] Jeremy L. Mineweaser, J. Scott Stadler, Susan Tsao and Maurice Glanagan, “Improving TCP/IP performance for the Land Mobile Satellite Channel”, MIT Lincon Laboratory, Lexington, pp. 711-718, IEEE 2001.

[19] D. B. Johnson, and D. A. Maltz, “Dynamic source routing in ad-hoc wireless networks”, in Mobile Computing, T. Imielinski and H. F. Korth, Eds., Norwell, M. A. Kluwer, pp. 153-191, 1996.

[20] K. Kant, “TCP offload performance for front-end servers”, Global Telecommunications Conference, Vol. 6, pp. 3242 – 3247, 2003.

[21] F. Khafizov, and M. Yavuz, “TCP over CDMA2000 networks”, Internet Draft, draft-khafizov-pilc-cdma 2000-00.txt.

[22] R. Ludwig, and R. H. Katz, “The Eifel Algorithm: Making TCP Robust Against Spurious Retransmissions”, in ACM Computer Communications Review, Vol. 30, No. 1, Jan. 2000.

[23] Marco Ughetti, Tiziana Trucco and Danilo Gotta, “Development of agent based peer-to-peer mobile applications on ANDROID with JADE”, Second International Conference on Mobile Ubiquitous Computing, Systems, Services and Technologies, pp. 287-294, IEEE Computer Society, 2008.

[24] V. D. Park, and M. S. Corson, “A highly adaptive distributed routing algorithm for mobile wireless networks”, in Proceedings of IEEE INFOCOM, Kobe, Japan, pp. 1405-1413, April 1997.

[25] C. Partridge, and T. J. Shepard, “TCP/IP Performance over Satellite Links”, IEEE Network Magazine, pp. 44-49, Sep.-Oct. 1997.

[26] Shu Lei, Wu Xiaoling, Xu Hui, Yang Jie, Jinsung Cho, and Sungyoung Lee: Connecting Heterogeneous Sensor Networks with IP based Wire/Wireless Networks, in the 4th IEEE Workshop on SEUSWCCIA' 06, pp. 2561-2566, 2006.

[27] Vinod Vulupala, and Vijay Kumar, “Dynamic Mobile IP and Nice-TCP for Improving TCP/IP Performance”, proceedings of the third International Conference on Mobile Data Management (MDM’02), IEEE Computer Society, 2002.

[28] Wooldridge, M., An introduction to multiagent systems, John Wiley Ed., 2002. ISBN 0-471-49691-X.

[29] Xiao Chen, and Jian Shen, “Reducing Connected Dominating Set Size with Multipoint Relays in Ad-hoc Wireless Networks”, Proceedings of the Seventh International Symposium on Parallel Architecture, Algorithms and Networks (ISPAN’04), IEEE Computer Society, 2004.

[30] Xiaohua Guo, Kougen Zheng, Yunhe Pan, and Zhaohui Wu, “A TCP/IP implementation for wireless sensor networks”, Proceedings of IEEE International Conference on Systems, Man and Cybermetrics, pp. 6081-6086, 2004.