

Multi Agent System based TCP for Wireless Networks

B. Sasikumar
Professor

Department of CSE
VINS Christian College of Engineering Nagercoil,
India

V. Vasudevan
Professor

Department of IT
A. K. College of Engineering Srivilliputhur, India

ABSTRACT

Due to the rapid evolution of networking and electronic capabilities, the use of wireless devices is becoming ubiquitous. Most of the wireless networks have fixed infrastructure and requires higher data rates. Due to high data rate in wireless networks the Transmission Control Protocol (TCP) performance has a broad and significant impact on data applications, and is essential towards massive deployment of service-providing agents and their widespread social acceptance. Wireless link losses result in poor TCP throughput since losses are professed as congestion by TCP. In order to increase the throughput, a Multi agent System (MAS) named Reusable Environment for Task-Structured Intelligent Networked Agents (RETSINA) has been introduced. This agent has the capability to reduce the impact of losses on TCP throughput and latency. In this work, we considered a video game application in which a networked virtual environment with high fidelity sound and video is created. Moreover, to efficiently use the video game application in a wireless environment, an Unreal Tournament Semi-Automated Force (UTSAF) multi agent has been proposed. UTSAF is a middleware written to take advantage of the power of gaming application systems by allowing them to participate in distributed simulations. The personal agent communicates wirelessly, in order to find the most appropriate option to serve the user. In addition, we comprehensively evaluated the performance of TCP with and without these multi agent platforms as technology choice and investigated its efficiency in a number of cases. We find that the simulations with the multi agent can reach the TCP throughput by as much as 95 % and the latency get reduced by 1-2 % as the simulation time steps in for every 100 seconds.

Keywords

Latency, Multi Agent System, RETSINA, TCP, Throughput, UTSAF, video game application, wireless network.

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.

Wireless networking is facing a growing need for efficient communication mechanisms that functionally integrate the varying and distinctive transmission characteristics. The end-to-end services provided by the TCP was designed and calibrated to overcome the problems related to stability, heterogeneity, fairness in bandwidth, efficiency in utilization

and congestion control. TCP is a reliable, connection-oriented, full-duplex, transport protocol widely used in wired networks [2]. TCP's flow and congestion control mechanisms are based on the assumption that packet loss is an indication of congestion. TCP is known to suffer performance degradation in mobile wireless environments. This is due to high bit error rates and mobility induced disconnections in real environments that are prone to packet losses.

TCP's performance is degraded as its congestion control mechanisms invoke inappropriately. The performance of TCP has been evaluated extensively under various scenarios over wireless links and efforts have been given to the impact of network channel losses. The losses are considered as congestion in the TCP and this result in source throttling and reduced throughput. Communication over wireless links is affected by high bit-error rate, temporary disconnections, high latencies and low bandwidth [20]. Losses due to wireless channel bit-error rate and mobility of the devices have a significant effect on the dynamics of transport layer protocols.

The wireless nodes are mounted on vehicles or mounted by human beings in multihop wireless networks to improve the performance of Medium Access Control (MAC) layer protocols and wireless routing protocols. Examples of multihop wireless networks are ad-hoc wireless networks, which has autonomous enabled finite number of radio equipped wireless nodes having an efficiently capable transmission radius to transmit message to all its surrounding neighbor nodes within the transmission range. In the route discovery of a source initiated on-demand routing protocol the flooding applications include paging a particular host or sending an alarm signal. In a one-to-all network the transmission of packets by a source node will reach all nodes that are within the predefined radius or range and in one-to-one network, each transmission is directed towards only one neighbor in that network [17].

In this paper, we concentrated our attention to the transport layer routing which uses the TCP. Transport connections in wireless networks are plagued by problems such as high bit rates, frequent route change and partitions. The throughput of the TCP over wireless networks is analyzed to be poor because the corresponding protocol considers the lost acknowledgements or delayed acknowledgements as congestion. The reason for poor performance in TCP has been examined in [5] and that need more effective mitigation strategies. The effect of channel conditions on TCP performance will provide robust, flexible protocol that consistently gives high performance for a variety of network environments. We proposed a new MAS to improve the performance of the TCP.

The rest of the paper is organized as follows. Section 2 reviewed about the related literature and section 3 describes the issues in TCP/IP. Implementation of multi-agent system is given in section 4. Section 5 details the experimental setup

and results analysis and finally conclusion and future scope is given in section 6.

2. RELATED WORK

In this section, we review the prior work on improving the TCP performance over wireless networks. Balan et al [3] suggested a TCP HACK, to distinguish between corruption and congestion losses and that preserves true end-to-end semantics. This enabled mobile node to check the received packet and determine whether the TCP payload or the TCP header is corrupted or not. If there is any corruption then actions are taken accordingly. Bellovin et al [4] studied the case if encryption is carried out using stream ciphers in the output feed-back mode or the cipher block chain mode. Such modes of stream ciphers are vulnerable to attacks and necessitate the use of authentication for integrity check. When integrity check is enabled, the corrupted packets are dropped by the IP-Sec protocol. Penchalaiah et al [14] proposed an approach that uses header compression to save bandwidth efficiently in adaptive nature. The proposed algorithm adjust the dimension of variable sliding window in W-LSB encoding with the accurate estimation of wireless channel state to achieve the good balance of compression ratio and error-resistant robustness for the adaptive use in wireless link. Riaz [16] made an analysis on TCP/IP over wireless broadband Internet, which can deliver up to 30 Mbps data in a 6 MHz channel.

Seshadri et al [19] proposed a router-assisted congestion control protocol called Quick Flow Control Protocol (QFCP). The QFCP can significantly shorten the flow completion time, fairly allocate bandwidth resource, and is robust to non-congestion related losses. Also an extensive study on quality of service for real time multimedia application is made and results showed an improved QoS for specified TCP/IP based scenarios. Cianca et al [8] developed an analytical framework to show the feasibility and effectiveness of truncated power control for higher TCP/IP transmission reliability. The proposed algorithm turns to be very effective when the time variability of the channel is high and improved the end-to-end throughput of a TCP connection. Cianca et al [7] presented a HAP-satellite integrated architecture for the provision of broadband connectivity at homes. HAP is exploited as intermediate node between the ground terminal and satellite. This approach improves the TCP/IP performance and its channel is more reliable. Radovanovic et al [15] proposed a protocol that takes advantage of application level framing and acknowledges missing transport layer segments with expired play-out deadlines. The performance of the protocol is compared with snoop, TCP RTM, and TCP Reno. Chan et al [6] presented a window regulator algorithm that uses the receiver window field in the acknowledgment packets to convey the instantaneous wireless channel conditions to the TCP source and an ack buffer to absorb the channel variations. The work improves the performance of TCP stack by 100 % over drop-tail algorithm for small buffer sizes in the congested router.

Mun et al [13] presented a window regulator algorithm that use the receiver window field in the acknowledgment (ACK) packets to convey the instantaneous wireless channel conditions to the TCP source and an ACK buffer to absorb the channel variations, thereby maximizing long-lived TCP performance. In-Su et al [11] proposed a lightweight TCP/IP stack called Compact Wireless-TCP/IP (CW-TCP/IP) to provide superior wireless performance for small embedded systems that have limited computing resources. This adopts TCP VenO as a congestion control algorithm to improve

wireless performance. Kumar et al [12] described a low cost wireless sensor network node, designed using ultra low power based MSP430 and Nordic nRF24L01. The concept has been implemented with biomedical wireless sensor network comprising of ECE sensor to communicate with the patients control device for monitoring and external communication. Senger et al [18] proposed a new network layer that transparently sits below the transport layer and hides non congestion-imposed packet losses from TCP. The network coding in this layer is based on the well-known class of Maximum Distance Separable (MDS) codes. Hongfu et al [10] presented a wireless communication method for water factory environment monitor system to send and receive wireless data by GPRS, and processes the data in sockets software, which setup both in server and client computers. Gopinath et al [9] proposed a Cache-and-Forward (CNF) protocol architecture to support efficient mobile content delivery services in the future Internet. The CNF architecture considers varying access link speed/quality and periods of disconnection as inherent properties of the network. Also a novel storage aware routing protocol is proposed to efficiently support mobile and wireless end-users through the use of a two-dimensional metric that takes into account both short-term and long-term path quality in making forwarding and storage decisions. Wang et al [22] proposed a novel network cognition and congestion control model based on Neuroendocrine-immune system that introduced natural inspired computation concept and is designed to improve the overall transmission performance of heterogeneous network.

3. TRANSMISSION CONTROL PROTOCOL

TCP was designed to connect multiple networks together in a seamless way. Transmission Control Protocol 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. 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 as TCP incorrectly responds to the losses 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 and which results in an unnecessary reduction of end-to-end throughput and its related performance [20].

However, the TCP is unable to distinguish a packet loss from a random loss; so, it considers any kind of loss as a sign of congestion. As a direct result, the occurrence of a random loss on a MAS link causes the TCP sender to invoke the usual congestion avoidance mechanisms. This leads to a drastic reduction of the transmission rate. While this conservative measure is appropriate to alleviate congestion, this has no effect on the occurrence of random losses; this reduces the link utilization and degrades the overall TCP performance, which is illustrated in figure 1.

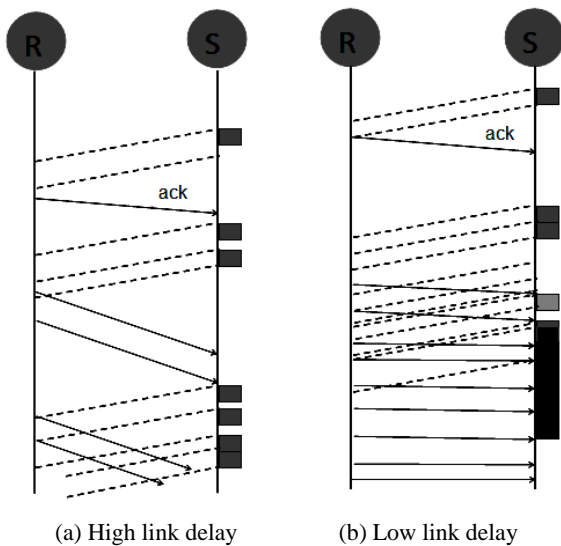


Figure 1 Link Utilization

4. PROPOSED ARCHITECTURES

The TCP provides modular congestion control, per-connection congestion control configurability, and the capability of automatically and intelligently changing the congestion control used by each connection according to its current environment. A new agent named RETSINA is supplemented with TCP to improve its performance by intelligent scheduling. This agent reduces the problems at the congestion window. When the agents execute tasks, they organize themselves to avoid processing bottlenecks and form teams to deal with dynamic changes in information, tasks, number of agents and their capabilities. Each individual RETSINA agent shall consist of agent reusable and reconfigurable components (i.e., a set of generic software components for knowledge representation, agent control and interaction with other agents). Each RETSINA agent consists of several modules, each of which is implemented as a multithreaded code. The main modules of a RETSINA agent are communicator, planner, scheduler, and execution monitor. The modules can operate asynchronously and concurrently.

4.1 Communicator

This provides an abstraction that supports peer-to-peer communication between agents based on the agent names. The communicator has Application Programming Interfaces (APIs) for other agent languages as well.

4.2 Planner

This receives goals through communication and finds alternative ways to fulfill them. The planning component is reusable and capable of accepting different planning algorithms.

4.3 Scheduler

This takes agent's current set of task structures as input, particularly the set of current executable actions, and schedules them.

4.4 Execution monitor

This takes the agent's next intended action and prepares, monitors, and completes its execution.

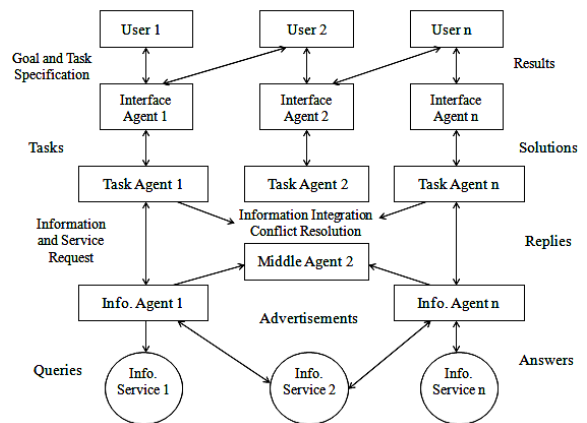


Figure 2 Multi-agent RETSINA System

Figure 2 shows an example of multi-agent RETSINA system in which each agent communicates with each other to perform a specific task. This system consists of three agents named interface agent, task agent and information agent. Interface agent interacts with users receiving their specifications and delivering results. Task agent formulate the plans and carry out them, which have knowledge of the task domain and which other task agents or information agents are relevant to perform various parts of the task. Figure 3 shows the overall architecture of the agent. The objective DB dynamically stores the objective with higher priority on the queue will be handled first by the planner. The task DB has data store unit that holds the plan. Tasks are added by the planner and removed by the scheduler. The schema library has a static data store unit that holds tasks schemas which are used by the planner for task instantiation. The task reduction library holds reductions of tasks and is used by the planner for task decomposition. The belief DB maintains the agent's knowledge of the domain in which the plan will be executed.

The proposed architecture was implemented with a gaming system by allowing to participate in distributed simulations with military video application. This requires high ended sound and video with high fidelity environment. So in order to improve the performance a bridging architecture named UTSAF is adopted for strategic planning, which involves different operators, agencies and partners to get better the interoperability [18]. The agent uses a multi domain, MAS in which each agent performs a specific task and interacts with other agents through a standard infrastructure to achieve interoperability between incompatible systems. UTSAF uses a modification route by the GameBots [1] to provide the simulation interface, which has two types of entities named human players who run individual copies of the game and connect to the server and bots, simulated players running simple reactive programs. This architecture uses a common communication infrastructure and language, agents can freely exchange information with one another and is extensible and able to support novel systems mediated by new agents. Figure 4 shows the UTSAF architecture a multi agent environment is used to span the gap between the two simulations. To keep the system scalable, diverse and extensible architecture allows subtasks to be assigned to specialized agents. A communication protocol between agents is essential to allow tasks and data to be distributed. To acquire information from the traffic, a Protocol Data Unit (PDU) parser extracts relevant information like entity type, location, velocity, and orientation from each unit is send to a SAF manager agent. SAF manager agents control the flow of information between OTBSAF and UT at the entity level. These agents receive

information from an SAF broker; update an internal database, and forward relevant updates to an agent representing the entity on the UT side of the simulation. GameBot agent serves

as the final connection between the agent space and the UT representation of the entities.

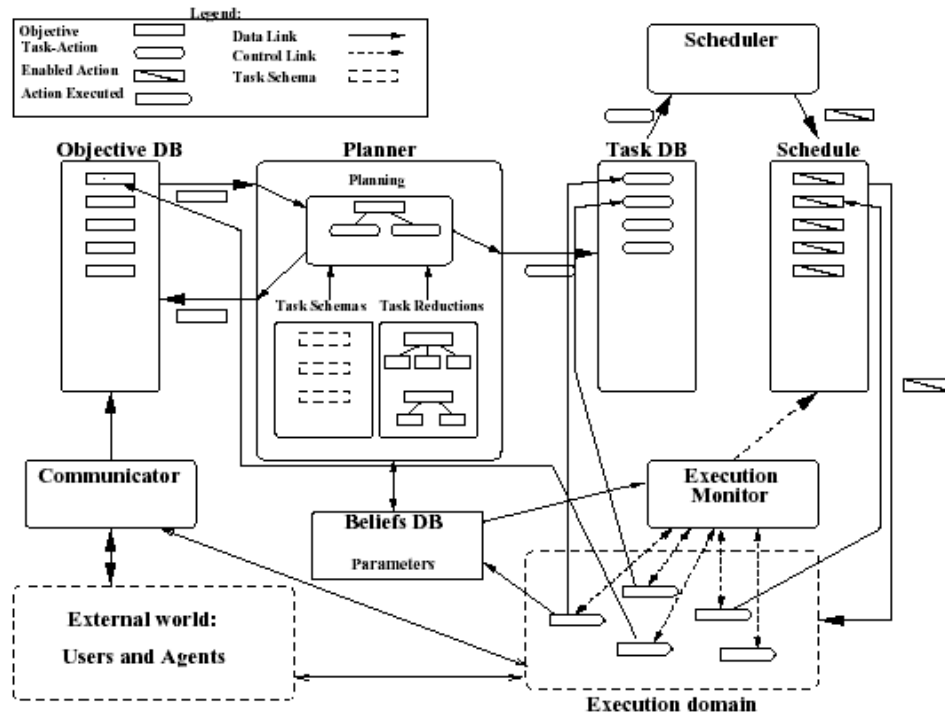


Figure 3 RETSINA Planning Architecture

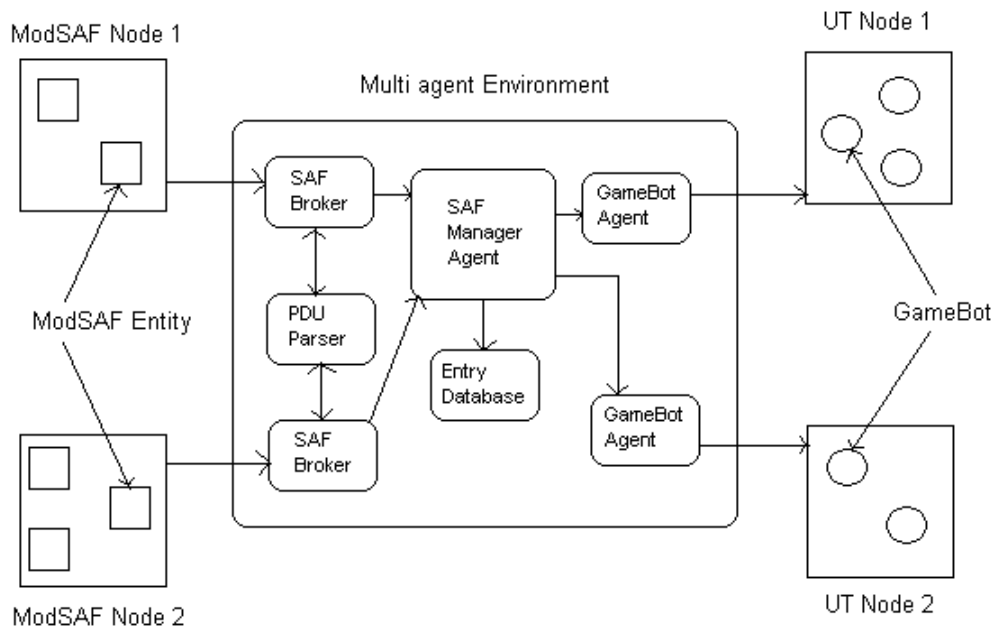


Figure 4 Multi-agent UTSAF System

5. SIMULATION RESULTS

All simulations are performed using ns-2, a network simulator that provides support for simulating wireless networks. The simulations were carried out on a wireless network environment consisting of 50 wireless mobile nodes roaming over a simulation area of 1000 meters x 1000 meters flat space operating for 900 seconds of simulation time. Each node has a radio propagation range of 250m and channel capacity was 2Mb/s. The TCP maximum window size is set to

500KB. We did our experiments with movement patterns for seven different pause times of 0, 100, 200, 300, 400, 500, and 600 seconds. The Protocol Data Unit (PDU) packets are generated at the rate of 100ms/packet and 20 ms/packet, respectively. The packet rate in the test bed is above 1000 packets/sec or more. The radio and IEEE 802.11 MAC layer models were used and the nodes move according to Random Waypoint mobility model.

Each simulation scenario uses a total of 16 TCP connections; each is 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. Each mobile node in the network starts its journey from a random location to a random destination with a randomly chosen speed. Once the destination is reached, another random destination is targeted after a pause by the mobile node. Once the node reaches the boundary area mentioned in the network, it chooses a period of time to remain stationary.

After the end of pause time, the node chooses a new direction, this time between 0 and 180 degrees, adjusted relative to the wall of the area on which the node is located. By varying the pause time, the relative speeds of the mobiles are affected. Performance of TCP/IP has been analyzed for the metrics throughput and latency. Unless otherwise noted, each one of the graphs presented in this section portrays one of these performance metrics on the y-axis as a function of speed and time on the x-axis respectively.

5.1. Throughput

Throughput is the amount of data that is delivered from the source node to the destination node in a network. Figure 5 shows the performance of throughput for the TCP with and without RETSINA agent. From the figure, it can be seen that, the amount of throughput gets reduced by 2 to 6 % when the speed is increased. We also see that, the performance of the proposed agent is 0.5 to 1 % better than the approach without the agent.

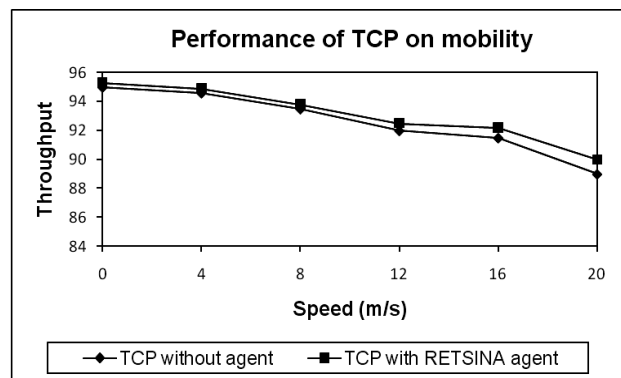


Figure 5 Analysis of throughput with RETSINA

Figure 6 shows the performance of throughput for the TCP with and without UTSAF during a military simulation video game application. It is observed that, the throughput linearly reduced by an average of 2 % when the mobility of the wireless node is varied from 0 to 20 m/s gradually. Moreover, the performance of the proposed agent is nearly 2 % better than the approach without the UTSAF architecture.

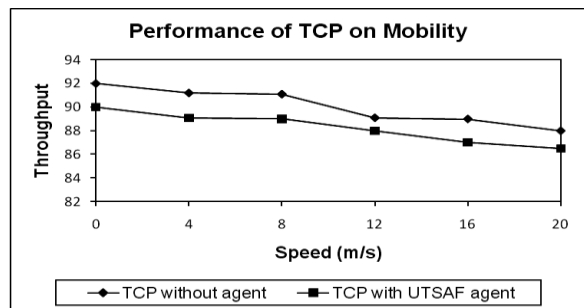


Figure 6 Analysis of throughput with UTSAF

5.2. Latency

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. Figure 7 shows the performance of latency for the TCP with and without RETSINA agent. It is seen that, the latency gets reduced by 5 % when the simulation time get on increasing. With the agent, the latency can be reduced by approximately 1 % throughout the simulation.

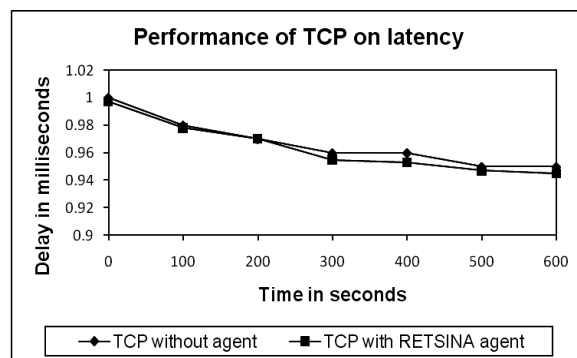


Figure 7 Analysis of latency for RETSINA

Figure 8 shows the performance of latency for the TCP with and without UTSAF. The figure pointed that, the latency is diminished by 10 % as the simulation is completed. With the UTSAF, the latency in the military simulated video game application can be reduced by 0 to 3 % throughout the entire simulation.

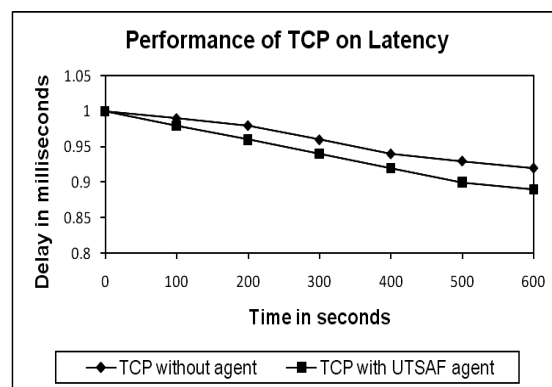


Figure 8 Analysis of latency for UTSAF

6. CONCLUSIONS AND FUTURE WORK

In this paper, we explored the multi-agent system with TCP to improve the performance of TCP in the wireless networks. The end-to-end throughput of TCP connection has increased while using the RETSINA agent. We performed the simulation by varying the speed for various scenarios observed that there is an average improvement of 2 %. Likewise the time delay for packet delivery has been reduced while using the multi-agent. All the simulation is also applied with a real time military simulated video game application in a wireless environment. This approach can easily be extended to integrate other components such as human behavior models, intelligent information gathering and filtering, and decision-making modules. Using agents to wrap incompatible systems lacking other provisions for interoperability allows the rapid construction of middleware. Using agents to wrap incompatible systems lacking other provisions for interoperability allows the rapid construction of middleware.

Finally we suggest that in future, transport protocol evaluations need to be more comprehensive. Evaluations should consider a range of realistic mobility models and should include special cases, such as high density and high traffic rates. Also the new performance metrics, such as energy based mobility, link stability, forwarding efficiency and mean delivery latency can be used to evaluate the proposed protocols. Simulations can also be evaluated by varying the mobility models in the simulation scenarios.

7. REFERENCES

- [1] Adobbati R., Marshall A. N., Scholer A., Tejada S., Kaminka G., Schaffer S., and Sollitto C., "GameBots: A 3D virtual world test-bed for multi-agent research" Proceedings of the second international workshop on Infrastructure for Agents, MAS, and Scalable MAS, May – June 2000, Montreal Canada.
- [2] Ajay Kr Singh, and Sridhar Iyer, "ATCP: Improving TCP performance over mobile wireless environments", Proceedings of the fourth International workshop on mobile and wireless communication networks, pp. 239-243, 2002.
- [3] Balan R. K., Lee B. P., Kumar K. R. R., Jacob L., Seah W. K. G., and Ananda A. L., "TCP HACK: TCP header checksum option to improve performance over lossy links", Proceedings of the IEEE INFOCOMM, April 2001.
- [4] Bellovin S., and Blaze M., "Cryptographic modes of operation for the Internet", Proceedings of the second NIST Workshop on Modes of Operation, 2001.
- [5] Bose P., Morin P., Stojmenovic I., and Urrutia J., "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] Chan M. C., and Ramjee R., "Improving TCP/IP Performance over Third Generation Wireless Networks", Proceedings of the twenty-third Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM 2004), Vol. 3, pp. 1893-1899, 7-11 March 2004.
- [7] Cianca E., De Luise A., De Sanctis M., Ruggieri M., and Prasad R., "TCP/IP Performance over Satellite and HAP Integrated Systems", Proceedings of the IEEE Aerospace Conference, Vol. 2, pp. 1209-1215, 6-13 March 2004.
- [8] Cianca E., Mauro De Sanctis Ruggieri M., and Prasad R., "Truncated Power Control for Improving TCP/IP Performance over CDMA Wireless Links", IEEE Transactions on Wireless Communications, Vol. 4, Issue 4, pp. 1704-1710, July 2005.
- [9] Gopinath S., Jain S., Makharia S., and Raychaudhuri D., "An experimental study of the Cache-and-Forward network architecture in multi-hop wireless scenarios", Proceedings of the 17th IEEE Workshop on Local and Metropolitan Area Networks (LANMAN), pp. 1, May 2010.
- [10] Hongfu Zhou, "Wireless Communication Design in Water Factory Monitor System by Sockets", Proceedings of the 4th International Conference on Wireless Communications, Networking and Mobile Computing (WiCOM), pp. 1, Oct. 2008.
- [11] In-Su Yoon, Sang-Hwa Chung, and Jeong-Soo Kim, "Implementation of Lightweight TCP/IP for small, Wireless Embedded Systems", Proceedings of the International Conference on Advanced Information Networking and Applications (AINA '09), pp. 965, May 2009.
- [12] Kumar V., Sonavane S. S., and Patil B. P., "Designing ultra low power Wireless Sensor Network with TCP/IP link", Proceedings of the 2nd International Conference on Adaptive Science and Technology (ICAST 2009), pp. 86, Jan. 2009.
- [13] Mun Choon Chan and Ramachandran Ramjee, "Improving TCP/IP Performance over Third-Generation Wireless Networks", IEEE Transaction on Mobile Computing, Vol. 7, Issue 4, pp. 430, April 2008.
- [14] Penchalaiah N., and Seshadri R., "Performance of TCP/IP/UDP adaptive header compression algorithm for wireless network", International Journal on Computer Science and Engineering, Vol. 02, No. 05, pp. 1646-1651, 2010.
- [15] Radovanovic I., Verhoeven T., and Lukkien J., "Improving TCP/IP Performance over Last-hop Wireless Networks for Streaming Video Delivery", Proceedings of the International Conference on Consumer Electronics (ICCE 2007), 10-14 Jan. 2007.
- [16] Riaz Ahamed S. S., "Performance analysis and special issues of wireless broadband systems in the information and communication environment", International Journal of Engineering Science and Technology, Vol. 2 (9), pp. 4533-4329, 2010.
- [17] Seddigh M., Solano Gonzalez J., and Stojmenovic I., "RNG and Internal Node Based Broadcasting Algorithms for Wireless One-to-One Networks", ACM Mobile Computing and Communication Rev. Vol. 5, No. 2 to appear.
- [18] Senger C., Schober S., and Tong Mao Zeh A., "End-to-End algebraic network coding for wireless TCP/IP networks", Proceedings of the IEEE 17th International Conference on Telecommunications (ICT), pp. 607, April 2010.
- [19] Seshadri R., and Penchalaiah N., "Improving quality of services from TCP/IP performance degradation", International Journal of Engineering and Technology, Vol. 3 (1), PP. 20-24, 2011.
- [20] Venkatesh Obanaik, Lillykutty Jacob, and A. L. Ananda, "Secure performance enhancing proxy: To ensure end-to-end security and enhance TCP performance over IPv6 wireless networks", International Journal on Computer Networks, Elsevier, (50), pp. 2225-2238, 2006.
- [21] 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.
- [22] Wang Hui, Liu Wei, Li Zheqing, and Fan Zhihui, "A novel network cognition and congestion control model based on NEI in next generation network", Proceedings of the 3rd International Conference on Computer Research and Development (ICCRD), Vol. 2, pp. 47, March 2011.