Volume: 6 Issue: 3 Pages: 2342-2346 (2014) ISSN : 0975-0290

Network Monitoring and Traffic Reduction using Multi-Agent Technology

M.Subramanian,
Assistant Professor, Department of Information Technology,
Scad College of Engineering and Technology, Tirunelveli – 414.
Email : m.subramanian86@gmail.com

ABSTRACT

In this paper the algorithms which could improve Transmission band and Network Traffic reduction for computer network has been shown. Problem solving is an area with which many Multiagent-based applications are concerned. Multiagent systems are computational systems in which several agents interact or work together to achieve some purposes. It includes distributed solutions to problems, solving distributed problems and distributed techniques for problem solving. Multiagent using for maximizing group performance with planning, execution, monitoring, communication and coordination. This paper also addresses some critical issues in developing Multi agent-based traffic control and monitoring systems, such as interoperability, flexibility, and extendibility. Finally, several future research directions toward the successful deployment of Multiagent technology in traffic control and monitoring systems are discussed.

Keywords - Agent, Data compression, Efficiency algorithm, Multiagent system, Transmission band,

Date of Submission: September 04, 2014 Date of Acceptance: November 13, 2014

I. INTRODUCTION

MULTI AGENT-BASED computing is one of the powerful technologies for the development of distributed complex systems. The reason for the growing success of agent technology in this area is that the inherent distribution allows for a natural network traffic and monitoring the system into multi agents that interact with each other to achieve a desired global goal. Recently, more and more agent-based traffic and transmission applications have been reported. Our literature survey shows that the techniques and methods resulting from the field of agent system and MAS have been applied to many aspects of reduce the traffic reduces and monitoring systems. This paper reviews multi agent applications in traffic reduction and transmission systems.

The network performance is being constantly improved and this results in better usage of the traffic control and transmission. There are several solutions to improve usage of a traffic control and transmission band:

- **Monitor the network and inform to the other agents,**
- **Increase the size of transmission band, advanced compression of data,**
- **Maximum transmission unit that can be transmitted one frame**.

The most significant influence on the behavior of OSI layers has the human and the application. Network Monitor is a network diagnostic tool that monitors local area networks and provides a graphical display of network statistics. Fig.1.

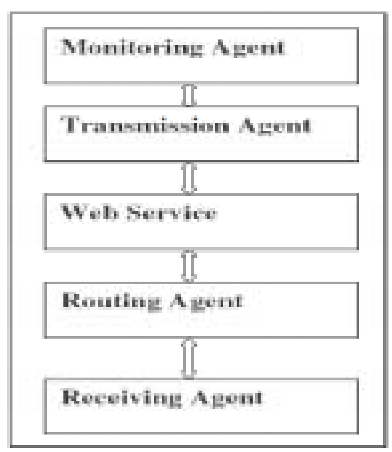


Fig. 1. **Multiagent System**

Human's behavior is one of the most important sources of data traffic in computer networks. The rest of the traffic resources (system and protocols) give relatively small contributions to this traffic. The network model should be extended.

Network administrators can use these statistics to perform routine trouble- shooting tasks, such as locating a server that is down, or that is receiving a disproportionate number of work requests. While collecting information

Volume: 6 Issue: 3 Pages: 2342-2346 (2014) ISSN : 0975-0290

from the network's data stream, Network Monitor displays the following types of information:

- **The source address of the computer that sent a frame onto the network. (This address is a unique hexadecimal (or base-16) number that identifies that computer on the network.)**
- **The destination address of the computer that received the frame.**
- **The protocols used to send the frame.**
- **The data, or a portion of the message being sent**.

Human behavior and user applications have to be considered. This allows better analysis of the transmission problem in computer networks. New issues considered in the model determine the working of OSI layers. For example, humans decide when and what will be sent through the network. Analyzing and controlling human's behavior can significantly improve efficiency and productivity of operation in computer networks.

II. BACKROUND

Network Traffic simulation is the principal approach for ITS application. The Multi-agent model is, in our opinion the most relevant computing model for network traffic simulation, allowing more accurate and realistic solutions behaviors to be simulated in s dynamic environment. Recently, a number of agent-based application related to its in different modes of transmission have been reported. Agent-based approach for network traffic and transmission band capacity allocation. In the next section, we present our multi-agent system for network traffic simulations. This system is based on network traffic reduction and multi agent technology which enhance the extensibility of the system.

III. MULTI AGENT-BASED TRAFFIC CONTROL AND MONITORING SYSTEMS

The operation of multi agents is supported and managed by distributed software platforms known as multi agent systems. The name of MASs usually refers to systems that support stationary agents, and web service systems support multi agents. A multi agent system provides mechanisms for agent management, agent communication, and agent directory maintenance. A mobile agent system provides additional mechanisms to support the migration and execution of multi agents. In an agent system, agencies are the major building blocks and are installed in each node of a networked system, in which agents reside and execute. To facilitate the interoperation of agents and agent systems across heterogeneous agent platforms, agencies designed to comply with agent standards are highly desired. Although more and more studies have been reported to apply agent approaches to traffic and transportation systems, only few researches address the system architecture and the platform issues of agent-based traffic control and management systems. An MAS to help traffic operators determine the best traffic strategies for dealing with network traffic and monitoring incidents. The multi agents in these two systems are implemented using the Java agent development Framework (JADE) agent platform [11], [12]. An developed a monitoring system and designed an multi agent-based real-time traffic control and management system. Multi agent is originally developed to enhance the distributed computing and information fusion capability for a traffic-detection system. Although it is a general purpose multi agent platform, Mobile-C is specifically designed for real-time and resource-constrained applications with interface to hardware.

Commonly used control architectures of multi agent based systems can be classified into hierarchical, hierarchical and hybrid. Generally, the hierarchical approach decomposes the overall system into small subsystems that have weak interaction with each other. On the other hand, the hierarchical approach is a completely decentralized approach in which agents communicate with each other to make independent decisions. Since the distributed agents only have a local view, it becomes difficult to predict the network state from a global perspective. The hybrid approach combines the features of hierarchical and hierarchical approaches. Most reported agent-based applications in traffic and transmission systems focus on developing MASs that consists of multiple distributed stationary agents. Mobile agent technology has not been widely applied in this area. To demonstrate the great value of mobile agents to intelligent transportation systems (ITSs), Chen et al. [9] integrate mobile agent technology with MASs to enhance the flexibility and adaptability of large-scale traffic control and management systems. Different from stationary agents, mobile agents are able to migrate from one host in a network to other hosts and resume execution in remote hosts. Multi agents can be created dynamically at runtime and dispatched to destination systems to perform tasks with most updated code and algorithms. Multi agent offers great opportunity to address challenges in traffic control and management, such as quick incident diagnosis, dynamic system configuration, new control-strategy deployment, and data-transmission reduction. To achieve flexible and intelligent control of traffic and transportation systems, Wang [4], [5] developed multi agent based networked traffic-management system. The multi agent-based control decomposes a sophisticated control algorithm into simple task-oriented multi agents that are distributed over a network. The ability of dynamically deploying and replacing control agents as needed allows the network to operate in a "control on demand" mode to adapt to various control scenarios.

The system architecture employs a three-level hierarchical architecture. The highest level performs reasoning and planning of task sequences for control agents. The multi agents are represented by monitoring agents that could migrate from remote traffic control centers to field traffic devices or from one field device to another. For the multi agent systems in network traffic management developed a test bed to allow designers of MASs to experiment with different strategies and examine the applicability of developed systems. The test bed consists of intelligent models for modeling intelligence of agents, a world model for representing traffic process, and an interaction model for modeling the interactions between multi agents. The communication in the test bed conforms to the FIPA standards.

IV. PACKING AND COMPRESSION USING MULTI AGENT

The "HTTP Optimizer" agent ystem is composed of an agent functioning on the server and an agent (client application) which emulates typical web browser operations.

The server-based agent (Server Agent class) has been implemented as a subclass of the default servlet (Default Servlet class), taking over its role, to assure a transparent operation of the system. In the case of a typical request, the agent uses the corresponding function of the Default Server class and sends a default response. Mode, the server-based agent packages all files from a WWW site directory and sends them in segments of size equal to or slightly smaller than 1500B (MTU: MSS + TCP/IP headers). Unlike a typical browser, one HTTP GET request concerning the HTML file is sent. For that reason, HTTP headers are to be found only in the first response package, thanks to which the MSS field can be used mainly for file content purposes.

• **Compressed packaging mode – a mode where, as well as packaging, compression concerning whole MSS segments is applied.**

Regardless of the mode used, the whole WWW site together with all its components is saved in a selected destination folder. The server-based agent carries out the process in the following way:

• **In the packaging mode, file data sent in a stream are separated by file size and name (Integer type). In this way, coming across an integer type value (4 bytes) and reading the file name (String), the client program can specify the place where the file data end (fig. 2)**

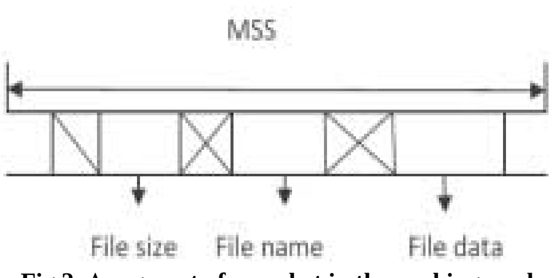


Fig.2. A **segment of a packet in the packing mode.**

In the compressed packaging mode, the process consists of two stages. First, data are formatted in the same way as in the packaging mode (size, filename, file data), and then the whole stream is compressed and put in the packet just after the compressed segment size.

In both cases it is the HTML file that is the first to be packaged, regardless of its size. Such assumption has been made for practical reasons so that the site can load even if its components have not been downloaded yet. All the packets (outgoing and incoming) concerning a particular connection are traced and displayed in a special program window to facilitate diagnostics of the preformed tests. The algorithm test (local) serves to perform the packaging algorithm tests on the local files. The packaging process is performed on the basis of the User-defined MSS value and the structure of each created packet is presented in the graph. 4 websites have been used for the tests, varying not only in components' size but also in their compression ratio.Fig.3.

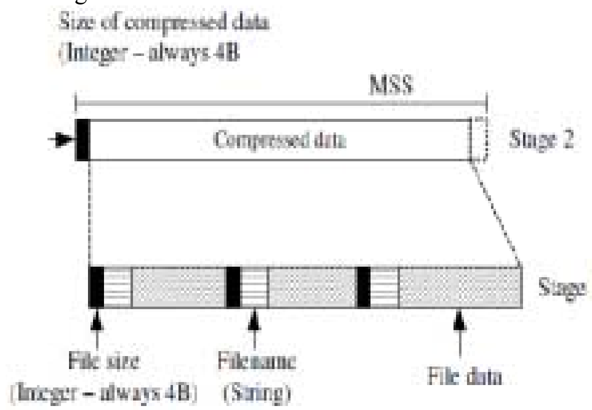


Fig.3. A Segment of a packet in the compressed packaging mode.

Each website has only one HTML file (index.html) which contains the descriptions and references to the images. The tested websites are:

Website1- a page containing very small images (icons) in the PNG format which are 200B in size on average. Those files have low compression ratio due to the PNG format which already employs lossless data compression. The compression ratios of all files do not exceed 1,1.

Website 2 – a page containing relatively big-sized photos (from 3KB to 20KB) in the JPEG format. The JPEG format employs lossy data compression which allows obtaining high compression ratios. The compression ratios of all files do not exceed 1, 2.

Website 3 – a page containing images of sizes 1.8 KB to 3.1 KB in the BMP format. The BMP files are usually not compressed (although this format provides lossless compression) and are typically of a large size. All images have a high compression ratio (1,3-1,9).

Website 4 – a page which is a combination of the above three pages. All files vary in terms of both size and compression ratio. Network tests, the results of which are presented below, have been performed in the local network to eliminate any negative factors concerning network congestion. In spite of that, after performing the tests in the global Internet network between computers separated by as many as 11 network nodes, the results were very similar **(Table 1).**

Types	Web1	Web2	Web3	Web4
Amounts of Packets - Web browser	**298**	**223**	**94**	**614**
Amount of Packets - Packing	**95**	**200**	**73**	**340**
Amount of Packets – Compressed packing	**46**	**184**	**42**	**253**
Coefficient E with standard transmission	**0.43**	**0.89**	**0.80**	**0.68**
Coefficient E with packing	**0.84**	**0.91**	**0.87**	**0.86**
Coefficient E with compression and packing	**2.19**	**0.98**	**2.40**	**1.21**

Table.1. Compression packets

The conception of "Compression" agent

In Agent technology information is being send much more frequently than when user is sending a request to website [12]. The text document (XML and JSON) can be compressed without losing any data. Packing a few documents in one like "HTTP optimizer" is impossible in most cases because the interaction between user and website can be lost. shows an example of website using agent technology with data compression.

Such requests cause the increase of traffic in network by sending small portions of information. The more often the user refreshes the website, the higher the increase. That ends in a decrease of user`s information in total transferred information. That is a disadvantage for the network efficiency. What makes AJAX more and more popular, is that user becomes more interactive with a task. Documents such XML or JSON (JavaScript Object Notation) are used to send data in asynchronous HTTP request. Therefore they can be easily compressed To estimate the efficiency (E) of implemented improvements we can calculate it for a given protocol as

$$E \quad —$$

Where D is the **data** volume to be transferred, and DT is the volume of total transferred information. The coefficient E satisfies the relation $E<1$ and $E=1$ for the ideal case (ideal protocol without compression) and $E>1$ (when compression is implemented).

Network traffic analysis

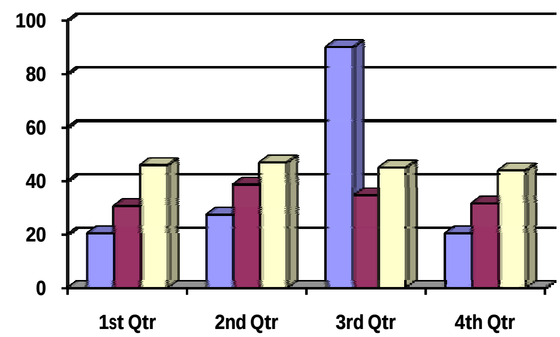
Before beginning with the tests themselves, we took some traffic captures with the purpose of finding the best way to distinguish the different uses that can be given to the protocol. We used several monitoring tools: tcpdump [7], ngrep [8] and wireshark [9]; these allowed us to analyze the packages in depth with help from their filters. Although this field was not present, it always existed in the captures we took. For the content-type field values MIME type values are used, according to RFC 2046, "Multipurpose Internet Mail Extensions (MIME) Part Two: Average Types". This RFC defines types of the xxx/yyy form, where xxx can be one of the following values: text, image, audio, video, application, message, multipart or HTTP; and yyy is a specialization of that.

Traffic Classification

Then, in order to be able to classify the traffic, it is necessary to have a tool that allows defining filters by content. Filter is a sort key for Net filter from Linux that identifies packages based on the Application layer data. This tool has predefined filters and allows the user to define new filters. The LL7 filters are regular expressions, which Were adapted to detect HTTP sessions with different content-type values.

V. EXPERIMENTAL RESULTS

In the experiment, simple two nodes network with client - server configuration has been used. Information was sent in three different ways: traditional, and then with proposed two algorithms. The obtained results were compared.



File size

Distribution in percentage of file sizes (smaller then Ethernet protocol MTU).

Traffic was measured by TCP dump, which is the linux tool (see www.tcdump.org tool for details). 1460 files were used with the total size of 785919 bytes. The average size was 538 bytes, with a distribution as on Fig. 7. It took 70 seconds to send unmodified information using FTP (File Transfer Protocol) and 21935 packets were required to transmit this information. When the algorithm 1 was used the 1460 files were packed to 565 packets, at a total size of 825339 bytes (average 1460 bytes). The number of files was reduced by to 38.7% (packets with data). In the case of algorithm 1 only 8498 packets were required and 28 seconds were needed to transmit the information. The number of sent packets had been reduced to 38.7%. These two numbers are identical, because the reduction of files and packets sent are proportional. In this experiment it is linear function, because collisions are eliminated. The transfer time was reduced to 40%.

Volume: 6 Issue: 3 Pages: 2342-2346 (2014) ISSN : 0975-0290

Tab. 2. **Coefficient of user data transmission efficiency and transfer time with algorithm 1 and algorithm 2.**

Transmission method	Data Transmission Efficiency	Time(s)	Packets
Standard	3.21	70	70
With algorithm 1	2.15	28	41
With algorithm 2	1.97	20	37

When the algorithm 2 was used the number of files was reduced to 394 data packets, at a total size of 579276 bytes (average 1472 bytes). Only 5930 packets were required and only 20 seconds were needed to transmit the information. The number of files as well as number of packets sent was reduced to 27%. The transfer time was reduced to 28.6%. Above results show an optimal usage of size MTU packet in a reduced amount of time, although the information sent was 5% bigger than the first one (because the added tags identified files in packets). Also, the number of sent packets had been reduced. The experiment contained two parts: preparing the packets and sending the information.

The algorithms were implemented in C++ language. In this case, the time to prepare the packets with algorithm I was 13 seconds and for second algorithm 17 seconds were required. In this stage of investigations we didn't optimize the speed of computer programs. We expect that after eliminating intermediate disk operations in these programs they will run at least 10 times faster. Anyway, in this experiment the total transmission times were reduced to 58.6% and to 52.9% respectively. Time needed to retrieve information on the receiving side (client side) can be neglected in comparison with time required for packet preparation. For the algorithm 1 (typically it was less than 10ms) it is only time required for localization file in unpacking packet and for algorithm 2 additional times for single compressing process of the data part of packet (typically was less than 100 ms).

VI. CONCLUSION

This paper has presented a method to measure the performance of network monitoring system operations using traffic reduction and multi agent approach. The initial results show that multi agent perform better when the number of managed agent increase significantly**.** In this paper, we have described techniques for compressing the link structure of the agent, and have preliminary experimental results on compression packet and access speed. The performed tests have proved that the developed multi agents system improves transmission band usage. As a result, the number of transmitted packets (frames) has been reduced several times.

VII. REFERENCES

[1]. M.Adler and M.Mitzenmacher,"Towards compressing web graphs", In Proc. of the IEEE DCC, March 2001, pp. 203-212.

[2]. T. Suel, J. Yuan, "Compressing the Graph Structure of the Web", Proc. of the IEEE DCC, March 2001, pp. 213-222.

[3]. R. Krashinsky, "Efficient Web Browsing for Mobile Clients using HTTP Compression", Technical Report MIT-LCS-TR-882, MIT Lab for Computer Science, Jan. 2003.

[4]. J. Nagle, "Congestion Control In TCP/IP Internetworks", RFC 896, 6 January 1984.

[5]. S. E. Spero, "Next Generation Hypertext Transport Protocol", InternetDraft, March 1995.

[6]. J. B. Postel, "Transmission Control Protocol", RFC 793, September 1981

[7]. B. Choi, N. Bharade, Network traffic reduction by hypertext compression, *Proc. of the International Conference on Internet Computing*, 2002, pp. 877-882.

[8]. E. Weigle, W. Fang, "A Comparison of TCP Automatic Tuning Techniques for Distributed Computing", *Proc. Of the 11th IEEE International Symposium on HPDC,* November 2002.

[9]. J. J. Garret, "Ajax: A New Approach to Web Applications",http://www.adaptivepath.com/ideas/essays/archives/000385.php, February 2005.

[10]. T. Bartczak, S. Paszczyński, B. M. Wilamowski: "Improvement the Computer Network Transmission Band Usage by Packing Agent Technology", *in Proc. of INDIN 2005 The 3rd International IEEE Conference on Industrial Informatics*, Perth, Australia, 10-12 August 2005, pp. 209-214.

[11]. J. Nagle, "Congestion Control In TCP/IP Internetworks", RFC 896, 6 January 1984.

[12]. S. E. Spero, "Next Generation Hypertext Transport Protocol", Internet Draft, March 1995.

[13]. J. Huang et al., "A Close Examination of Performance and Layer Characteristics of 4G LTE Networks," *Proc. 10th Int'l Conf. Mobile Systems, Applications, and Services* (MobiSys 12), ACM, 2012, pp. 225– 238.

[14]. S. Dey, "Cloud Mobile Media: Opportunities, Challenges, and Directions", *Proc. Int'l. Conf. Computing,Network and Commun.*, 2012, pp. 929–33.

[15]. K. Lu, Y. Qian, and H.-H. Chen, "A Secure and Service- Oriented Network Control Framework for WiMAX Networks," *IEEE Commun.Mag.*, vol. 45, no. 5, May 2009,pp. 124–30.

[16]. M. Schlansker, Y. Turner, J. Tourrilhes, and A. Karp, "Ensemble Routing for datacenter networks," in *Proc. 2010 ACM/IEEE Symposium on Architectures for Networking & Commns Systems*

[17]. S. Bitam and A. Mellouk, "ITS-cloud: Cloud computing for intelligent transportation system," in Proc. IEEE Global Network. Conf., Anaheim, CA, USA, 2012, pp. 2054–2059.

[18]. Y. Zhang, B. Chen, and X. Lu, "Intelligent monitoring system on Refrigerator trucks based on the internet of things," Wireless Communications and Applications. Berlin, Germany: Springer-Verlag, 2012, pp. 201–206.