

Cohesive Token Passing Algorithm Utilizing Software Agents

A. Fua'ad Abdulrazzak and Shamala Subramaniam
Department of Communication Technology and Networks,
Faculty of Computer Science and Information Technology,
University Putra Malaysia, 43400 Serdang, Selangor, Malaysia

Abstract: Problem statement: The communications domain has utilized the implementation of protocols for a wide spectrum of applications. This encompasses Medium Access Control (MAC) protocols. MAC protocols have been extensively researched from several angles. This encompasses the implementation in the area of Wave Division Multiplexing (WDM) networks and Mobile Adhoc Networks (MANET). The relevance of intelligence in sustaining the pre-requisites for dynamic reconfiguration has gained an integral attention in MANET. **Approach:** The implementation of Token Ring in MANET can be correlated to its complementary implementation in IP networks. In this paper, the limitation of Token Ring algorithm for IP networks in the context of intelligent processing has been researched extensively. An enhanced Token Ring protocol governed by intelligent processing has been implemented in this paper. The core of the new protocol is based on the circulation mechanism of the token. As opposed to the traditional circulatory mechanism, a software agent is designed to become an intelligent circulatory agent in this research. The developed software agent is utilized to implement prioritized token access subject to the traffic type. Each station is coupled with a software agent who cohesively collaborates to assign the token. **Results:** The proposed agent and the enhanced Token Ring implementation have been extensively verified through simulation experiments. A complete circulation of the ring is defined upon all nodes being visited at least once. Discrete-event simulation models were developed and deployed for the purpose of performance analysis. The results acquired validated the improved results of the new software agent based implementation. The performance metrics studied were average delay and average buffer utilization. **Conclusion:** The proposed algorithm has enabled to derive an ideal balance between the complexity of intelligent processing and the versatility of managing the token ring.

Key words: Software agents, cohesive medium access control, discrete-event simulation models

INTRODUCTION

In token-passing ring local area networks, the shared transmission medium is closed and it has a formation of loop. The information transmission is unidirectional. Access to the transmission medium is regulated with the help of a token-small packet that consists of about eight bits. All stations are attached to the transmission medium using active interfaces that can modify the information passing through it. When the station has some information to transmit, its interface monitors the transmission medium and searches for immediately a free token. As it detects a free token it captures the token, changes the status of the token from idle to busy. Subsequently the data is transmitted immediately (Nasro *et al.*, 2008). Token

ring protocol has been an active research area in both wired and wireless networks (Malpani *et al.*, 2004). Much of these researches have focused on the design of routing and Medium Access Control protocols (MAC), since efficiency of these protocols have a direct and significant impact on the performance. MAC protocols provide access arbitration for the network nodes to the transmission medium. This research developed an agent based token ring protocol that utilizes agent intelligence to provide an efficient routing mechanism to enhance and increase the network performance. Token passing protocols have been widely utilized. It is more suitable for the industrial environment because it is both efficient and deterministic at high loads. However, at low channel traffic, it does not perform as well as the contention protocols. This lack of efficiency becomes

Corresponding Author: A. Fua'ad Abdulrazzak, Department of Communication Technology and Networks,
Faculty of Computer Science and Information Technology University Putra Malaysia, 43400 Serdang,
Selangor, Malaysia Tel: +60389466577 Fax: +60389466512

more serious as the number of nodes and the (bandwidth) \times (distance) product are increased. Moreover, occasional losses of the explicit token result in inaccessibility periods that increase the maximum packet delay calculated in the steady state of the network.

The token circulation mechanism is an important factor of the token passing protocol efficiency. Many algorithms have been introduced in order to utilize the network topology information and changed the token circulation mechanism, with ensuring the total order of the message delivery among the nodes. To have an idea, Let us consider what would happen if the token circulation algorithm were to ignore the network topology and choose an arbitrary order to visit the nodes. For instance, in a ring consisting of nodes 1, 3, 5, 2, 6, 4, 1, if the nodes are visited in the order 1; 2; 3; 4; 5; 6; 1; 2; 3; 4; 5; 6; :::, then between any two consecutive visits, the token takes several hops-in this case, although the length of the round is 6 (which is optimal), message overhead is large (since each node visit requires the token to take several hops). On the other hand, visiting the nodes in the order 1, 3, 5, 2, 6, 4, 1, 3, 5, 2, 6, 4... still results in the optimal length, but lower message overhead. Thus, the latter visit order should be referred. However, if the visit order is chosen without taking the topology into account, in general, the algorithm will not typically choose the best possible order of visits.

The above example suggests that it is useful to utilize network topology information in determining the order in which nodes are visited (Lu *et al.*, 2006). In this study, software agents have been integrated and developed to enhance the token ring protocol using intelligence agent technique enables. The developed software agent is utilized to implement prioritized token access subject to the traffic type.

Related work: The concept of token passing can be traced to the early work of Newhall, Pierce and others in the late 1960s, wherein a control signal or token is generated to continuously loop throughout a ring or bus network. The purpose of the token is to identify who has control over the connecting medium and to transfer this control at regulated times to ensure equal access. As the token travels along its path, it is subsequently retrieved, regenerated and returned to the connecting medium by each and every node on the network. This process of repeating the signal is accomplished for all signals transmitted on the network, making the nodes a vital part of the communication process. To transmit a message, a node removes a circulating token, modifies it into a data packet and places it again on the network

for transmission to its destination. Once the recipient node receives the data packet, an acknowledgment is sent in the modified token to the sending node, which converts the packet into the original token and returns the token to the network. As each node can only hold a token for a specified duration, the repetition of this cycle results in a deterministic, predictable access to the medium for all node on the network (Prakash and Baldoni, 1998).

The token passing protocol and the token circulation mechanism have been studied and analyzed from different aspects in different type of networks. These analyses cover the challenges facing these networks in terms of their performance and efficiency. The IEEE 802.5, token access mechanism is defined as follows. A token is circulated around the ring. Whenever a station has data to transmit, it waits for the token arrival. Upon receiving a token, it can transmit for a fixed interval called the Token Holding Time (THT). After the transmission, the station either releases the token immediately or after the arrival of all the frames it transmitted. Using this scheme, a station on an station ring may have to wait as long as an $n \times$ THT interval to receive a token. This may be unacceptable for some applications if n or THT is large. For example, for voice traffic and real-time applications, this interval may be limited to the 10-20 ms range. Using the token access method severely limits the number of stations on the rings. The timed-token access method solves this problem by ensuring that all stations on the ring agree to a 'target' token rotation time TTRT and limit their transmissions to meet this target as much as possible (Grow, 1982). Here, TTRT. If the THT is positive, the station can transmit for this interval. At the end of transmission, it releases the token. If a station does not use the entire THT allowed, other stations on the ring can use the remaining time by using the same algorithm.

In WDM optical fiber networks, the token ring protocol has been integrated using the piggybacking token ring protocol (Selvakennedy and Ramani, 1999).

This is due to the bottleneck problem that may appear because of the large capacity difference between the fiber optics and the electronic processing of the nodes and also the need to utilize the multi channels provided by the WDM optical networks. In the piggybacking protocol, when a node receives the token it senses all the channels simultaneously, if two or more channels are sensed idle then the token is transmitted on one channel and data are transmitted on the other, if only one channel available then the token is transmitted on this channel followed by the data, which is piggybacked onto the token. In MANET the totally ordered message delivery is achieved by continually

circulating a token through all the nodes of the network in a virtual ring, the token circulates around the virtual ring carrying a sequence number. When a node receives the token, it assigns sequence numbers (carried with the token) to its messages and then multicasts the messages to the group members. The sequence number carried in the token is incremented once (by one) for each message sent by the node holding the token. Since the messages are assigned globally unique sequence numbers, total order can be achieved. (Additional mechanisms are needed depending on the desired level of reliability (Malpani *et al.*, 2004). An algorithm that circulates a software agent to collect information about the nodes and network topology is introduced for ad hoc networks (Choudhury *et al.*, 2000). The procedure used by agents to travel through the network is analogous to such algorithms described above which changes the token circularity. However, the deployment of agents in the IEEE 802.5 has not been explored.

MATERIALS AND METHODS

A large number of factors influence the decision to use any particular scientific technique to solve or analyze a given problem. A system can be simplified to such an extent that it can be solved analytically. Such an analytical solution is desirable because it leads to a close form solution, where the relationship between the variables is explicit. However, such a simplified form of the system is obtained by making several assumptions so as to make the solution mathematically tractable. Most real-life systems are so complex that some simplifying functions are not justifiable. Simulation imitates the behavior of the system over times and periods data as if the real system were being observed.

Network model: In order to analyze and compare the performance of the enhanced token ring protocol based on software agent, a C++ discrete event-driven simulator was implemented. Event scheduling approach has been used in this simulation. Packet arrival generation rate is based on Poisson distribution using a C++ built in random number generator rand() which returns a sequence of apparently non-related numbers each time (Stroustrup, 1997).

Assumptions: The following assumptions were considered:

- Stations are heterogeneous
- Arrivals at all stations follow a Poisson process

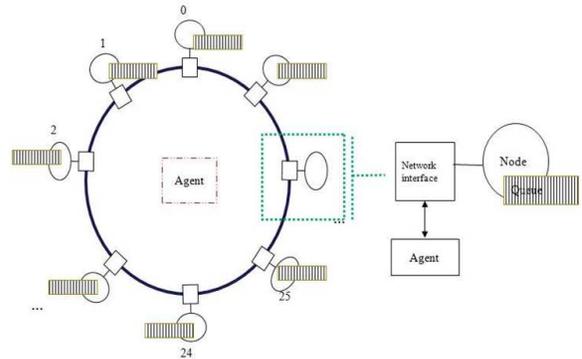


Fig. 1: Agent based token ring network model

- Even stations generate real time traffic while odd stations generate non real time traffic. The transmission medium is considered to be error free
- The spacing between the stations is the same
- The source and destination stations on the average half the ring part
- The propagation delay is about 5 μ sec per kilometer of transmission medium. Network size is 50 nodes. Packet generation rate for $\lambda C\{30, 60, \dots, 270\}$

Simulation Model development: As shown in Fig. 1, the model used in this paper consists of N number of sources producing different traffic (video as a real time traffic and text as non real time traffic). It is assumed that even nodes generate the real time traffic while odd nodes generate non real time traffic. The token circulation and packet scheduling is implemented depending on the developed software agent technique. Three main events have been considered in this model, which is arrival of packet, departure of packet and token. When the node receive the token two case have to be considered, if the node has a real time data then it serve this data and use the software agent to determine the next node has a real time data in the network and just pass the token to the first node satisfy this condition, if no real time data in any node in current round then it will pass the token to the next node has an arrival data. The second case is if the node has a non real time then it will use its agent to determine wether there is another a new real time date have been arrived in any node so the priority of passing the token to this token is given, if not then it will serve its data and then the agent again will be used to determine the next node. If there are no packets arrived in any node then the current node

holding the token will disable passing the token until it gets information from its agent about the next node the token passed to. An important application in this model is ensuring the total order of message delivery in the network. When using a token been visited at least once considering also the direction circulation algorithm, a round is said to be complete when every node have of passing the token (one direction). This model is described to meet the QoS requirements for the real time traffic as well as providing better buffer management utilization for it.

To clarify the basic idea of the Software Agent function in the enhanced token ring protocol, Fig. 2 shows a Pseudo-code illustrating how the agent works.

Performance metrics: The performance metrics studied were average delay and average buffer utilization. The selection of these two metrics came due to their important impact on the network efficiency. As discussed above, the lack of efficiency in token ring networks becomes more critical as the number of nodes and the (bandwidth) × (distance) product are increased because that will maximize the packet delay in the node buffer as will, as the buffer utilization will be affected also due to the continuous buffered packets. The average delay was defined as the time, on the average, for the packet takes to travel from the sending to the receiving node, with the networks' queuing model. However the buffer utilization was defined as the percentage of the buffer capacity that is in use at any particular time or for a duration of time.

Variables in *token*:

Visited []: array of Booleans, one entry per node, indicates whether that node has been visited yet in the current round; initially all false

```

1  When node i receives token
2  token.visited[i] := true
3  next := getNext()
4  send token to next
5  function getNext() returns node id
6  N: set of ids of all nodes

      To show the idea of the agent function
7  UV: = {k: token.visited[k] = false}
8  UVRN: = {unvisited real time nodes}
9  If (UV = 0) then
      // all nodes are visited and the round is completed
10 Token.Visited[k] := false, 0 < k < n
11 return 0
12 else if (UVRN = 1) then
      // there is unvisited real time nodes
13 return the RT node that has a request
14 else
      // no more real time but there is an unvisited non-real time nodes
15 return any such k
16 endif
17 endif
    
```

Fig. 2: Agent based token ring algorithm

RESULTS AND DISCUSSION

Two different scenarios are analyzed. The first compares the original study average delay (for both real time and non-real time traffic) with the average delay resulted from the proposed Software Agent technique in token ring protocol. Second scenario compares the results of buffer utilization in both works.

The results analyzed are delay (subject to the traffic sensitivity) and the buffer management utilization. In Fig. 3 the result of the average delay against packet generation rate for the token ring network is given to show the normal average delay and the dynamic traffic is introduced and we can see that the average delay for both traffics are the same and equal to the average delay/2 that is because half of the station generate a real time packets (e.g., video) and the other half generate non-real time traffic (e.g., text) without any priority given to any traffic type. In Fig. 4 after introducing the proposed model of the software agent that produces a significant improvement as compared to the normal token ring model. The enhancement achieved ranges from 49% in the case of light load and decreases to 7% in the worst case of the heavy load.

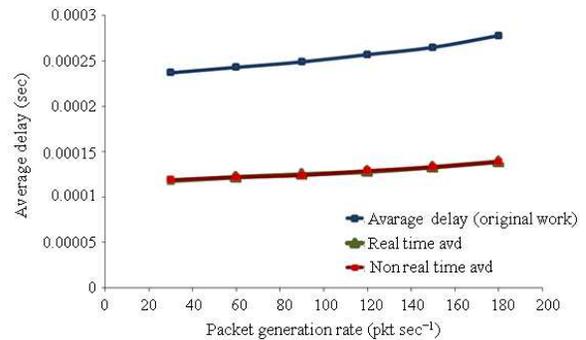


Fig. 3: Real time and non real time delay Vs average delay

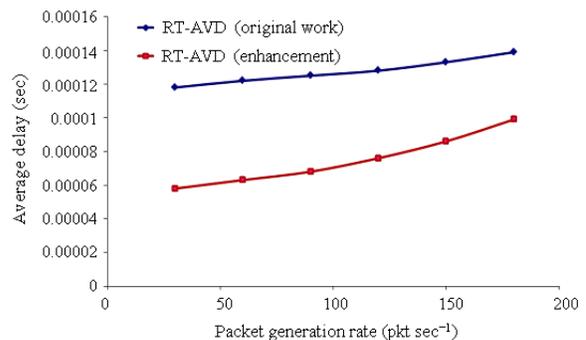


Fig. 4: Real time delay Vs (applying SA teaching)

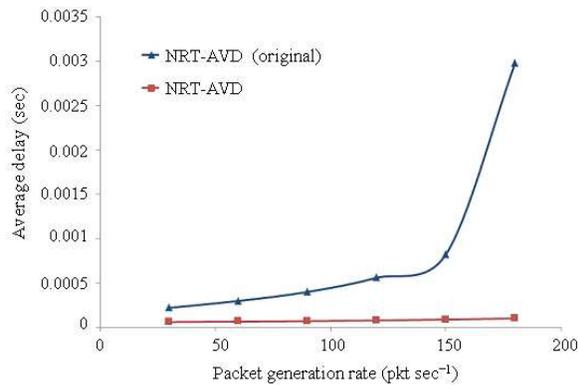


Fig. 5: Non-real time delay Vs (applying SA teaching)

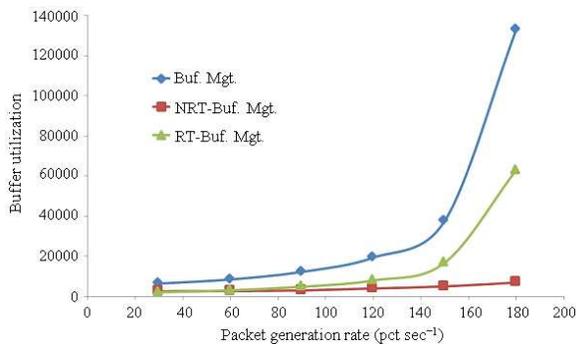


Fig. 6: Buffer utilization for RT and NRT traffic Vs original work (applying SA teaching)

Figure 5 shows that the non-real time traffic suffers in terms of performance. This is due to giving the priority to the real time traffic. The fairness of serving the different type of traffic can be introduced as a new research issue in next stages of this research.

Figure 6 shows the clear improvement achieved on the buffer utilization in the real time nodes and shows that the buffer utilization in non-real traffic nodes still close to the average buffer utilization of the network.

CONCLUSION

Software agent is an efficient technique in communication networks area. Agents have significantly made an impact on a number of technologies. It has been applied to aspects of artificial intelligence research, expert systems, genetic algorithms, distributed processing, distributed algorithms, collaborative online social environments and security.

In this study the proposed agent-based algorithm provides an effective and intelligent token ring

circulation based on agent that enhances the performance of the network and maintains the QoS constraints for the real-time traffic.

The results acquired through the simulation of using agent proved that the delay performance of the network and the average buffer utilization has been improved and the QoS for sensitive traffic is ensured.

The proposed future work can be summarized as follows:

- Introducing a threshold in the real time traffic to ensure the fairness and service grantee for different types of traffic
- Studying multi-traffic nodes instead of different types of nodes (real time and non-real time)
- The queuing system can be studied also by introducing new techniques like defining multiple queues for multiple traffic types
- Implementing this idea using an agent based simulation language like JAVADEMOS, AGLETS.

REFERENCES

Choudhury, R., S. Bandyopadhyay and K. Paul, 2000. A distributed mechanism for topology discovery in ad hoc wireless networks using mobile agents. Proceedings of the 1st ACM International Symposium on Mobile Ad Hoc Networking and Computing, (MANC'00), IEEE Press, Boston, Massachusetts, pp: 145-146. <http://portal.acm.org/citation.cfm?id=514151.514179>

Grow, R.M., 1982. A timed token protocol for local area networks. United States Patent 4459588. <http://www.freepatentsonline.com/4459588.html>

Lu, X., G. Fan, and R. Hao, 2006. A dynamic token passing MAC protocol for mobile ad hoc networks. Proceedings of the 2006 international Conference on Wireless Communications and Mobile Computing, July 03-06, ACM Press, Vancouver, British Columbia, Canada, pp: 743-748. DOI: 10.1145/1143549.1143698

Malpani, N. Y. Chen, N.H. Vaidya and J.L. Welch, 2004. Distributed token circulation in mobile ad hoc networks. IEEE Trans. Mobile Comput., 4: 154-165. DOI: 10.1109/TMC.2005.25

Nasro, M.A., E. Manzoor, J. Xing and Y. Wang, 2008. Enhancing feasibility analysis of IEEE 802.5 token ring. Proceeding of the International Conference on Wireless and Optical Communications Networks-The Next Generation Internet, May 2008, IEEE Xplore Press, Indonesia, pp: 1-5. DOI: 10.1109/WOCN.2008.4542510

- Prakash, R. and R. Baldoni, 1998. Architecture for group communication in mobile systems. Proceeding of the IEEE Symposium on Reliable Distributed Systems, Oct. 1998, IEEE Xplore Press, Purdue University, Lafayette, pp: 235-242. DOI: 10.1109/RELDIS.1998.740505
- Selvakennedy, S. and A.K. Ramani, 1999. Performance study of token-passing protocol for traffic multiplicity in optical networks. Proceeding of the IEEE International Conference on Parallel Processing, Sept. 21-24, IEEE Xplore Press, USA., pp: 524-531. DOI: 10.1109/ICPP.1999.797441
- Stroustrup, B., 1997. The C++ Programming Language. 3rd Edn., Addison-Wesley, ISBN: 0-201-88954-4, pp: 685.