ADIAN: A Distributed Intelligent Ad-Hoc Network

Saeed Shahbazi[1], Gholamreza Ghassem-Sani[2], Hamidreza Rabiee[2], Mohammad Ghanbari[3], and Mehdi Dehghan[4]

[1] Iran Telecommunication Research Center, Tehran, Iran
shahbazi@ce.sharif.edu
[2] Computer Engineering Department, Sharif University of Technology, Tehran, Iran
{sani, rabiee}@sharif.edu
[3] Department of Electronic Systems Engineering, University of Essex, United Kingdom
ghan@essex.ac.uk
[4] Computer Engineering Department, Amirkabir University of Technology, Tehran, Iran
dehghan@ce.aut.ac.ir

Abstract. Mobile Ad-hoc Networks are networks that have a dynamic topology without any fixed infrastructure. To transmit information in ad-hoc networks, we need robust protocols that can cope with constant changes in the network topology. The known routing protocols for mobile ad-hoc networks can be classified in two major categories: proactive routing protocols and reactive routing protocols. Proactive routing protocols keep the routes up-to-date to reduce delay in real-time applications but they have high control overhead. The control overhead in reactive routing protocols is much less than proactive routing protocols; however, the routes are discovered on demand, which is not suitable for real-time applications. In this paper, we have introduced a new routing system for mobile ad-hoc networks called ADIAN, which is based on the concepts of Distributed Artificial Intelligence (DAI). In ADIAN, every node acts as an independent and autonomous agent that collaborates with other agents in the system. Our experimental results have verified the efficiency and robustness of ADIAN under dynamic conditions of ad-hoc networks.

Keywords: Mobile Ad-hoc Networks, Routing, and Distributed Artificial Intelligence.

1 Introduction

A Mobile Ad-hoc Network (MANET) is a network consisting of wireless devices that make a self-configured network together. There is no fixed communication infrastructure in MANETs. Since wireless devices' broadcasting range is limited, communication in MANETs depends on the intermediate nodes. Therefore, each node in the network acts as a router. In these networks the topology changes constantly due to the mobility of the nodes. Furthermore, new nodes may be added to the network, existing nodes may leave the network, or some nodes may go to the sleep mode, dynamically. Due to special characteristics of these networks, the main problem is how to setup an

S. Chaudhuri et al. (Eds.): ICDCN 2006, LNCS 4308, pp. 27 – 39, 2006.

effective routing mechanism to deliver data packets. Another problem involves power consumption. Since most of the nodes are battery operated and have limited power, power consumption of nodes should be minimized.

Distributed nature and dynamicity of MANETs, make it suitable for applying Distributed Artificial Intelligence (DAI) techniques. Many routing protocols have already been designed for MANETs; however, most of them use simple assump-tions to model the mobility of the nodes. What is more, they have a modular structure containing object codes. Intelligent agents embody stronger notion of autonomy than objects. Likewise, the agents are capable of flexible behavior, and the standard object model has nothing to say about such types of behavior. Moreover, cooperation of a set of autonomous and intelligent agents can tolerate and handle potential failures in MANETs.

In this paper we introduce a new routing system called A Distributed Intelligent Ad-hoc Network (ADIAN), where network nodes are considered as intelligent agents and the agents discover routes to deliver information. Agents in ADIAN are autonomous and act in a plausible way. In this system, the routing overhead, which has an important impact on the performance of the MANETs is aimed to be minimized. This point is discussed further in section 4-3.

The remainder of the paper is organized as follows: sections 2 and 3 briefly present the existing routing protocols of two major categories: proactive and reactive routing. The weakness of the existing routing protocols and the motivation for using DAI techniques is then discussed in the section 3. Section 4 includes a detailed description of ADIAN. In section 5 the simulation results are presented. Finally, section 6 concludes the paper.

2 Routing in Ad-Hoc Networks

Dynamic nature of a MANET due to the mobility of its nodes causes a high degree of unpredictability in the network topology. This unpredictability makes the task of routing for transfer of information very complex. Design of a robust routing algorithm in these networks is an important and active research topic. Various routing protocols have already been introduced and evaluated in different environments and traffic conditions [5-6]. An extensive review and comparison of routing protocols for MANETs can be found in [7].

Environment and features of the MANETs, such as mobility and limited energy and bandwidth, requires an efficient use of available resources. In other words, to preserve the power, the routing overheads should be minimized and routing loops need to be avoided. Other important issues include: scalability, directional link support, security, reliability, and QoS [13-15].

The existing routing protocols of MANETs are divided into two major categories: 1) proactive routing protocols and 2) reactive routing protocols [5]. Proactive routing protocols constantly keep the routes between each pair of nodes up-to-date, by using periodic broadcasts. Since routing information is kept in some routing tables, these protocols are sometimes called table-driven protocols. On the other hand, reactive routing protocols discover a route only when it is required. Moreover, discovery process of a route is often initiated by the source node [5, 14].

The main feature of the proactive routing protocols is to maintain fixed routes to every pair of node in the network. Creating and maintaining routes are performed through periodic and event-driven messages (such as triggered messages when a link is broken) [5, 14]. Some of the proactive routing protocols include: DSDV[1], CGSR[2], WRP[3], TBRPF[4], and FSR[5] [16-18, 21-24].

In reactive routing protocols, in order to reduce the routing overhead, routes are discovered only when they are needed. Some of the reactive routing protocols include: DSR[6], AODV[7], TORA[8], ABR[9], and SSR[10] [25-28].

3 Application of DAI in Routing in Ad-Hoc Networks

Distributed and dynamic nature of the MANETs makes this domain suitable for applying DAI techniques. Most of the ad-hoc routing protocols do not use these techniques. Since there are a few routing protocols that use artificial intelligence, having a new system in which the intelligent agents can collaborate for routing has motivated us to apply some of the DAI techniques to the routing problem in the ad-hoc networks. Multi-agent systems offer production systems that are decentralized rather than centralized, emergent rather than planned, and concurrent rather than sequential.

In this section we briefly explain two routing protocols that have used simple DAI techniques. Moreover, there are some related works that try to use artificial intelligence techniques in [29-32], but they do not exploit autonomous agents in their algorithms.

3.1 ARAMA

ARAMA is based on the concepts in biology [1]. The idea of designing ARAMA is based on the Ant Colony. Forward packets (Forward Ants) are used to collect information and backward packets (Backward Ants) are used to update the routing information in the nodes. Motivation of the algorithm is based on similarity between the MANETs and the ant routing algorithm (i.e., both of them have similar features such as their self-built, self-configured, and distributed nature). Some of the advantages of this algorithm are: fast response to the changes, local solution, employing both of

1 Destination Sequenced Distance Vector.
2 Cluster-head Gateway Switch Routing.
3 Wireless Routing Protocol.
4 Topology Dissemination Based on Reverse-Path Forwarding.
5 Fisheye State Routing.
6 Dynamic Source Routing.
7 Ad hoc On-demand Distance Vector.
8 Temporally Ordered Routing Algorithm.
9 Associatively Based Routing.
10 Signal Stability Routing.

reactive and proactive advantages, discovering multiple routes, reliable routes, ability to control the updates and the broadcasts.

3.2 Ant-AODV

One of the disadvantages of AODV is the lack of ability to handle real-time applications. Moreover, the ant type routing algorithms can not work well in highly dynamic networks with weak routes. Since nodes are dependent on ants for collecting information, in some cases the nodes carrying ants may leave the network unpredictably. This is caused by nodes mobility, and sleep mode of the mobile hosts. In this case, the number of ants in the network is decreased, which results in ineffective routing [2].

Ant-AODV is designed to solve the existing weaknesses of AODV and the ant routing. Some of its characteristics are: decreasing end-to-end and route discovery delay. Unlike other routing protocols, it does not waste bandwidth used for routing overhead, either.

Ant-AODV and ARAMA use swarm intelligence as one of the DAI methods. In the swarm intelligence, each agent can not solve the problem or even part of it alone. In other words, these protocols do not act autonomously and do not have the ability to make decision in various domains and different situations independently.

Therefore, none of the existing routing protocols for MANETs is suitable for all the conditions. In other words, each protocol is usually designed for a special purpose and for a special domain.

In the rest of this paper we will focus on a new routing system called ADIAN, which has been implemented as a framework to test different conditions. We can test various situations in topology and size of the network in order to determine the important criteria in robustness of the network.

4 ADIAN

ADIAN discovers the routes on-demand and is based on nodes negotiation as intelligent agents. Agents act autonomously in ADIAN. Routing in ADIAN is based on agents' negotiation to deliver data packets. The negotiation protocol between agents ADIAN is to some extend similar to that of CNET [9]. Moreover, each node has uncertain and limited knowledge about the agents in other nodes, which are represented in a way similar to meta-knowledge of MINDS algorithm [10]. Finally, the routing process is achieved through cooperating agents. Furthermore, the agent's knowledge is updated through negotiation with other agents, and by data packets' transmission.

4.1 Knowledge Store in ADIAN

In ADIAN, each agent stores its knowledge in four tables that consist of: State Table, Routing Table, Neighborhood Table, and Belief Table. The "State Table" includes the latest agent's information. The "Routing Table" contains the latest information about the destination agents and the appropriate neighbor agents that are used to deliver data packets to the desired destination. The "Neighborhood Table" includes a list of neighbors, which is updated periodically. The subsection 4.2 explains how these tables are used in ADIAN.

The Belief Table contains information about every node, which is accompanied by a belief degree about the accuracy of the information, and an updating time of each record. There are some other fields in this table including: Destination Agent, Belief Degree, Position of the agent, Remained Power, and the other important resources such as: CPU Load, Congestion Level, and whether or not the agent is busy. This table is used to choose the best neighbor to negotiate for delivering data packets to a desired destination.

Each agent in ADIAN learns the status of other agents through communication. Whenever a new agent enters into the system, it will construct its own belief table. The information in this table is later updated based on the information of the received data packets from other agents. At the start point of adding a new record, the belief degree value is set to the value of sender's belief degree. Then, through a punishment/reward mechanism, the degree of each agent belief, which indicates the accuracy of its meta-knowledge, gradually converges to a stable state.

The position information of nodes in ADIAN is assumed to be supplied by a GPS. Note that the information of each agent is local and no agent has a full view of the whole system. In order to know the position of other agents, each agent has to rely on its meta-knowledge about others.

In Figure 1, the typical knowledge of the first agent about the second agent is represented. This knowledge-base shows that agent 1 is 80% certain about the accuracy of its information about agent 2. This uncertainty gradually reduces through negotiation between the agents, and by transmission of data packets through the network.

Belief Table												
ID	Source Node	Destin. Node	belief	Power	pos X	pos Y	Transmission Delay	Band Width	Congestion Level	CPU Load	Is Busy	Time
1	agent1	agent2	80%	178	5	1	0.1ms	128K B	2%	18%	No	27

Fig. 1. Format of the Belief Table – each agent has knowledge about the other agents with a belief degree

4.2 Routing in ADIAN

The ADIAN routing includes the three following phases.

1) Route Discovery

The new routes are found in this phase. Agents are responsible for delivery of data packets from a source node to a destination node, while trying to find an optimal route. Therefore, they go through a negotiation process to find a suitable route to deliver the data packets. If an agent intends to deliver a packet, and it does not find any neighbor in its routing table, it will search its neighborhood table based on its own information about the destination that exist in its Belief Table to choose the best neighbor. The selection is based on the following factors: Euclidean Distance, Belief Degree, Remained Power, and Updating Time using the following equation:

$$\text{Distance} = \frac{\text{Euclidean Distance}}{\frac{1}{3}\left(\alpha \frac{updatingtime}{currenttime} + \beta * belief + \delta * power\right)} \quad (1)$$

Where α represents the importance of updating time in choosing a route, β is the belief's importance level, and δ is the remained power importance degree.

The effectiveness of α, β, and δ is discussed in section 5.2, and simulation results compare their effectiveness.

During route discovery, the agent that has the least distance is chosen to negotiate. Then a message is sent to the chosen agent for cooperating in transmission of the data packets. This agent evaluates the received information as indicated below.

(a) Information about the destination accompanied with the belief degree of the source agent regarding that information.

(b) Some other information such as delivery priority of the data packets used to increase the system performance.

Agents in ADIAN can accept/reject the negotiation autonomously. In some cases, if there is any congestion or limited resource in the selected agent, it will deny to cooperate in routing that ends in having a balanced network and a robust routing.

As an example, consider Figure 2, where agent A decides to send some data packets to agent E. It assesses its neighbors, and uses its knowledge about E that is found in its Belief Table to calculate the distances between itself and its neighbors, using the above equation. In this example, the least distance belongs to the agent B; therefore, it is chosen for negotiation.

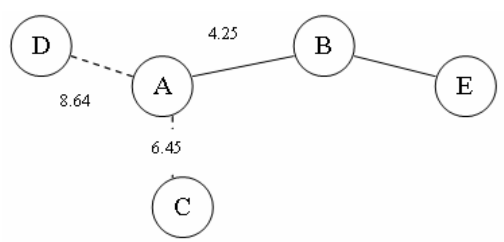

Fig. 2. Topology of a sample ad-hoc network – agent A is trying to send its packets to agent E

If B has enough power to deliver packets, it will evaluate the received information for inconsistencies. If there is any inconsistency between its knowledge and the received information, it will return the correct information to A, and will not accept the negotiation. Agent A will then correct its knowledge. Otherwise, it will send a message in order to inform A that it can cooperate with it.

In ADIAN, data packets are also used to update information that agents have about each other. In other words, each data packet carries some extra information regarding a limited number of lastly visited agents. Besides, if an agent accepts to cooperate with another agent, it will send back its information to the requested agent. The initial performance of ADIAN might be low due to local and inaccurate knowledge of its agents. However, this exchange of information facilitates the gradual convergence of ADIAN to a stable and acceptable performance.

As it was explained earlier, the knowledge of ADIAN agents is gradually refined via their communications. The performance of ADIAN is highly dependent on the accuracy of its agents' knowledge. Therefore, in the cases where only few agents communicate, the system might not have a reasonable performance. To handle this problem, a periodic broadcasting scheme is used. To reduce the overhead, the broadcasting is done only if an agent remains idle for a certain period of time.

2) Route Maintenance

The second phase of routing is the route maintenance which is responsible for maintaining the routes during the transmission task. In ADIAN, there is no need to send additional control packet to maintain routes. Data packets in their journey are used to update the knowledge of the visited agents; therefore, agents receive up-to-date information about each other.

Another problem is to prevent routing loops. ADIAN prevents routing loops using a list of illegal neighbors. If a data packet passes through a node, then it adds the previously visited node to its list. Agents are not allowed to use the nodes in their illegal list for routing.

3) Failure Handling

The third phase of routing is about handling the potential failures, which are often due to mobile nodes, and sometimes are due to having low battery in nodes that contribute to a transmission task. If a link between two nodes fails, the related information in their routing table will be deleted and the current agent tries to negotiate with another agent. If there is no suitable agent for routing, a backward routing will be performed to the previously visited agent.

4.3 ADIAN Features

ADIAN satisfies the following requirements that have been specified in [19] such as:

- The process of routing need to be performed cooperatively.
- Routing loops should be prevented.
- Routing should be initiated on-demand.
- The possibility of having a *sleep mode* need to be considered (i.e., when the power of agents are less than a threshold, they deny cooperation in delivering packets).
- Agents’ knowledge about the world and one another is local, limited, and uncertain.
- In distributing the tasks through the network, the load balancing should be considered.
- The routing algorithm should be complete (i.e., if there exists a route to a destination, the algorithm would find it).

In addition, there is no need to use control packets in ADIAN, so that the bandwidth is preserved and the ADIAN Overhead is low.

5 Simulation Results

To assess the performance of the proposed protocol, that consists of a Multi Agent Systems (MAS), called ADIAN, we have used various simulations in a typical mobile Ad-hoc network environment, as described below.

5.1 Simulation Model

In order to demonstrate the effectiveness of ADIAN, we evaluate our proposed protocol and compare its performance to the DSR and AODV. We have implemented ADIAN using the GLObal MObile SIMulation (GLOMOSIM) library [33].

The number of the nodes in the simulation world is 40; however, the size of the simulation world including the number of nodes, and the mobility pattern of nodes could be simply configured by adjusting the simulation parameters. In addition, system parameters such as available power, updating time, and beliefs in choosing the next hop are all configurable, and have been tested in different states.

The details of the ADIAN's simulation model, including the transmission primitives, mobility and traffic model, are reported next.

5.1.1 Transmission Primitives

Here, an ideal scheduler controls the packet transmissions and each agent uses a FIFO buffer. The size of the buffer is limited to 20 packets. A broadcast packet is initiated after the channel is free for a Random Assessment Delay (RAD) randomly chosen in the range [0, 1, 2, ..., 250] milliseconds (ms) with a transmission radius of R = 250 meter. By notifying a packet reception to the contractor (i.e. the selected sender's neighbor for delivering data packets) about remaining for the whole duration of the transmission within the transmission range, no collisions with other transmissions would be occurred, simultaneously.

The required time to detect a link breakage is simulated by considering a typical retransmission mechanism with an exponential back-off. The nominal transmission speed is set to 8 Mbps. This simplified model would present the main behavior of a typical wireless link layer in our simulations.

5.1.2 Traffic

Packets are generated by 20 Constant Bit Rate (CBR) sources at the rate of 5 packets/sec. The size of data packets is different and is chosen from a source file. The parameters in the source file records are: "time", "source agent", "destination agent", and "data packet size", where each field is separated by a special delimiter. Note that the scenario for the simulation is fixed. However, we can change the scenario easily. The number of simultaneous sources at any time can be more than one by defining more sources that have the same time field value.

5.1.3 Mobility

Agents can move in an n * m kilometer (km) region according to the random waypoint mobility model with a zero pause time.

The default values were set to n = m = 1 km. At the beginning of simulations the agents were placed randomly inside the region. Each agent then selects a new point and moves towards it at a constant speed, that is chosen in the range of [1, 2, …, V_{max}] m/s, uniformly at random. When the agent arrives at the destination, it would repeat the same behavior.

5.1.4 Performance Metrics

The following metrics have been considered during the simulations.

- *Delivery Rate*; ratio of the number of data packets delivered to the destinations generated by the traffic sources.
- *Time Cost*; lasting time to deliver data packets to the destination.
- *Physical Distance*; Euclidean distance among hops.

5.2 Comparison Between Different Scenarios

In the simulations, the cost includes temporal and physical distance. The temporal cost is related to the negotiations (1 per negotiation) and sending the packets to the next hop (2 per sending). For calculating the physical distance, the Euclidean distance is used. The simulation model is similar to the one described in section 5.1. The pause time for each agent to settle down and then move was set to 100 seconds.

In this section, different experiments have been performed to determine the desired coefficients of the system parameters. In the simulations, we have evaluated three parameters including: delivery rate, time cost, and physical distance cost. Here, the agents move randomly such that some links maybe formed to increase performance, or vise versa. In other words, a suitable topology maybe formed, that increases the performance, or conversely, an undesired link may occur, which decreases the performance. To minimize the effect of this phenomenon, we have forced the desired pattern by statistical analysis.

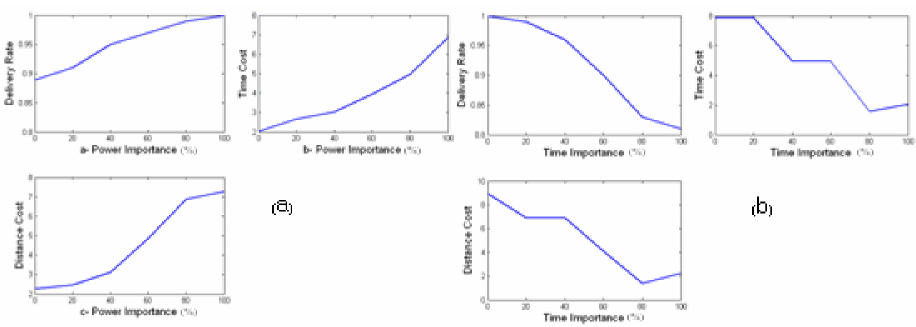

Fig. 3. (a) The effect of importance of power to choose best next hop: a- Delivery rate (%), b- Time Cost (10ms), c- Physical distance (20m), (b) The effect of importance of updating time to choose the best next hop: a- Delivery rate (%), b- Time Cost (10ms), c- Physical distance (20m)

Figure 3(a) shows that when the level of importance of power choosing the next hop increases, the delivery rate will also increase. In other words, the probability of having no power for each node, which leads to go to sleep mode, will decrease. This fact is shown in Figure 3(a)-a. According to Figures 3(a)-b and 3(a)-c, there is a trade-off between increasing delivery rate and increasing the costs. By using the results of these experiments one can determine the desired parameters for the desired performance for different applications.

In Figure 3(b), the importance of updating time factor is shown. When the updated time factor is given more importance the effects of power and the delivery rate will decrease (Figure 3(b)-a). According to the Figures 3(b)-b and 3(b)-c, the costs decrease by increasing the importance of the updating time. This is because, more up-to-date routes means less backtracks in the routing process. Therefore, agents need to have fewer negotiations.

Figure 4 shows the result of experiments where the degree of agents' belief regarding other agents was gradually given a higher priority. Similar to the updated time effect, by increasing the importance of belief degree, the importance of the power decreases. Therefore, the delivery rate will decrease. This fact is shown in Figure 4(a). According to Figures 4(b) and 4(c), the costs decrease by increasing the importance of the belief degree, which is due to the fact that transmissions face less deadlock (i.e., paths and cooperating agents are chosen more accurately).

Our experimental results showed that the power and the belief degree factors have more effects on the system than the updating time.

We have also performed some experiments regarding the levels of information carried with the data packets, as shown in Table 1. In the first set of experiments, which is shown in row 1 of the table, there is no additional information in the data packets. The results of carrying information about 1, 2, and 3 last visited nodes are shown in rows 1, 2, and 3 of the table, respectively. According to Table 1, increasing the number of visited nodes in data packets that carry information will increase the performance.

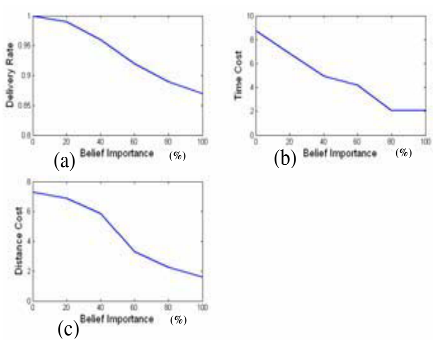

Table 1. Experimental results of effects of added information to data packets on system performance

# of visited nodes that data packets carrying their information	Distance Cost (×20m)	Temporal Cost (×10ms)	Delivery Rate (0-1)
0	4.97	7.28	0.8
1	3.9	3.16	0.9
2	3.04	3.13	0.96
3	2.05	2.27	0.99

Fig. 4. The effects of importance of belief degree to choose the best next hop: (a) Delivery rate (%), (b) Time Cost (10ms), (c) Physical distance (20m)

5.3 Comparison with Other Protocols

To show the effectiveness of ADIAN, comparisons with two typical proactive (DSDV) and reactive (AODV) routing protocols were made. The first one compares the overhead of the protocols, and the other one is about the measurement of

"GOODPUT". The conditions of the simulation remain the same, as expressed in section 5.1. Furthermore, the coefficients of ADIAN in section 5.2 are set to 33.33%.

Figure 5(a) shows the comparison results of ADIAN, DSDV, and AODV Delivery Rate. The results indicate that ADIAN has the best performance in correctly delivering the data packets. This is due to the agents' negotiations to find a path to the destination, as was explained earlier in section 4.2.

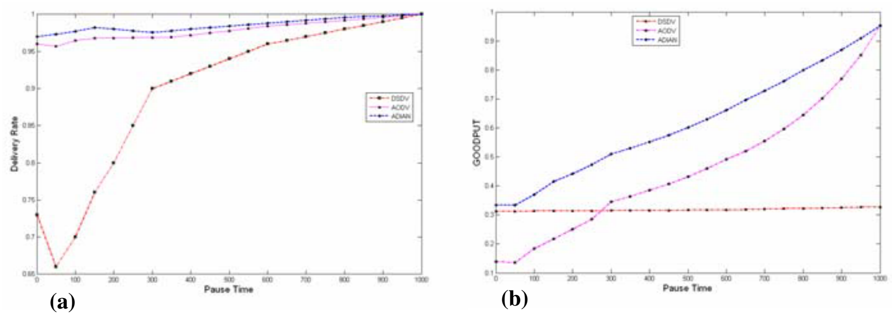

Fig. 5. (a) Delivery rate comparison of ADIAN, AODV, and DSDV. (b) GOODPUT comparison of ADIAN, AODV, and DSDV.

The comparison of "GOODPUT" factor is shown in Figure 5(b). This is a measure that shows the probability or the rate of successfully received packets with no cell loss that causes packet loss at the receiver. The results show that the GOODPUT factor of ADIAN is the best. This is the consequence of ADIAN's low overhead that was discussed earlier, in section 4.2.

6 Conclusions

Ad-hoc networks are flexible networks that do not have any pre-installed infrastructure. By recent developments in wireless technology and peripheral devices, the application of such networks has been rapidly increased. However, the routing problem in ad-hoc networks due to mobility of nodes is still a challenging issue.

In this paper, we have presented a new routing system called ADIAN, for ad-hoc networks. In ADIAN, routing is performed by the help of DAI methods, and each node in the network is regarded as an autonomous agent. Therefore, we have achieved to design a robust routing algorithm by using intelligent agents. Moreover, we have been successful to decrease the routing overhead.

In this paper, the simulation results were based on various parameters such as the life of power supply, update time, and agents' belief about other agents. The results show that, in different conditions, ADIAN gradually converges to the desired point of operation by minimizing the costs and the resource consumptions. This gradual convergence is due to inaccurate knowledge of its distributed agents in the early stages of the routing algorithm.

References

[1] O. Hussein, T. Saadawi, "***Ant Routing Algorithm for Mobile Ad-hoc networks (ARAMA)***" Source: IEEE International Performance, Computing and Communications Conference, Proceedings, 2003, p 281-290.

[2] Shivanajay Marwaha, Chen Khong Tham, Dipti Srinivasan, ***"Mobile Agents based Routing Protocol for Mobile Ad- hoc Networks***", IEEE Global Telecommunications Conference, v 1, 2002, p 163-167.

[3] Belding-Royer, Elizabeth M., Perkins-Charles, "*Evolution and future directions of the ad hoc on-demand distance-vector routing protocol*". Source: Ad Hoc Networks, v 1, n 1, July, 2003, p 125-150.

[4] Gerhard Weiss, "***Multiagent Systems A Modern Approach to Distributed Artificial Intelligence***", MIT 1999.

[5] E.M. Belding-Royer, C.-K. Toh, "***A review of current routing protocols for Ad hoc mobile wireless networks***", IEEE Personal Communications Magazine (April 1999) v 6, n 2, p 46–55.

[6] Jean-Pierre Ebert, Brian Burns, Adam Wolisz, "***A trace based approach for determining the energy consumption of a WLAN network interface***", Proceedings of European Wireless, February 2002, pp. 230–236.

[7] C.E. Perkins, "***Ad hoc Networking***", Addison-Wesley, Reading, MA, 2000.

[8] Alex L. G. Hayzelden, John Bigham, "***Software Agents for Future Communication Systems***", Springer, 1999.

[9] Randall Davis, Reid G. Smith, "***Negotiation as a Metaphor for Distributed Problem Solving***", Artif. Intell., vol 20, pp. 63-109, 1983.

[10] Uttam Mukhopadhyay, Larry M. Stephens, Michael N. Huhns, Ronald D. Bonnell, "***An intelligent system for document retrieval in distributed office environments***", Journal of the American Society for Information Science, Vol. 37, Issue 3 (May 1986), pp. 123-135.

[11] Alex L. G. Hayzelden, Rachel A. Bourne, "***Agent Technology for Communication Infrastructures***" Wiley, 2001.

[12] Ryokichi Onishi, Saneyasu Yamaguchi, Hiroaki Morino, Hitochi Aida, Tadao Saito, "***The Multi-agent System for Dynamic Network Routing***", IEICE Transactions on Communications, v E84-B, n 10, October, 2001, p 2721-2728.

[13] R. Beraldi, R. Baldoni, "***Unicast routing techniques for mobile Ad hoc networks***", in: M. Ilyas (Ed.), Handbook of Ad hoc Networks, CRC Press, New York, 2003 (Chapter 7).

[14] Elizabeth Belding-Royer, "***Routing approaches in mobile Ad hoc networks***", in: S. Basagni, M. Conti, S. Giordano, I. Stojmenovic (Eds.), Ad hoc Networking, IEEE Press Wiley, New York, 2003.

[15] M.S. Corson, J. Macker, "***Mobile Ad hoc networking (MANET): routing protocol performance issues and evaluation considerations***", RFC 2501, IETF, January 1999.

[16] C.E. Perkins, P. Bhagwat, "***Highly dynamic destination sequenced distance-vector routing (DSDV) for mobile computers***", Computer Communications Review (October 1994) 234–244.

[17] C.-C. Chiang, H.K. Wu, W. Liu, M. Gerla, "***Routing in clustered multi hop, mobile wireless networks with fading channel***", in: Proceedings of IEEE SICON_97, April 1997, pp. 197–211.

[18] S. Murthy, J.J. Garcia-Luna-Aceves, "***An efficient routing protocol for wireless networks***", ACM Mobile Networks and Applications (MONET) Journal, Special Issue on Routing in Mobile Communication Networks, October 1996, pp. 183–197.

[19] Joseph P. Macker and M. Scott Corson. "***Mobile Ad hoc networking and the IETF***". Mobile Computing and Communications Review, 2(1):9.14, 1998.

[20] B. Bellur, R.G. Ogier, F.L. Templin, "***Topology broadcast based on reverse-path forwarding (TBRPF)*** ", IETF Internet Draft, draft-ietf-manet-tbrpf-01.txt, March 2001.

[21] "***APE: Ad hoc Protocol Evaluation test bed***", Department of Computer Systems at Uppsala, Sweden., ACM/Kluwer Mobile Networks and Applications Journal 6 (3) (2001) 211–222.

[22] L. Kleinrock, K. Stevens, "***Fisheye: a lens like computer display transformation***", Technical Report, UCLA, Computer Science Department, 1971.

[23] P.F. Tsuchiya, "***The Landmark Hierarchy: a new hierarchy for routing in very large networks***", Computer Communications Review 18 (4) (1988) 35–42.

[24] G. Pei, M. Gerla, X. Hong, "***LANMAR: landmark routing for large scale wireless Ad hoc networks with group mobility***", in: Proceedings of IEEE/ACM MobiHOC 2000, Boston, MA, August 2000, pp. 11–18.

[25] D.B. Johnson, D.A. Maltz, "***Dynamic source routing in Ad hoc wireless networks***", in: T. Imielinski, H. Korth (Eds.), Mobile Computing, Kluwer Academic Publishers, Dordrecht, 1996, pp. 153–181.

[26] C.E. Perkins, E.M. Royer, "***Ad hoc on-demand distance vector routing***", in: Proceedings of 2nd IEEE Workshop on Mobile Computing Systems and Applications, February 1999.

[27] V.D. Park, M.S. Corson, "***A highly adaptive distributed routing algorithm for mobile wireless networks***", in: Proceedings of INFOCOM _97, April 1997.

[28] R. Dube, C. Rais, K.-Y. Wang, S. Tripathi, "***Signal stability based adaptive routing for Ad hoc mobile networks***", IEEE Personal Communications, February 1997, pp. 36–45.

[29] Zheng, Xiangquan; Guo, Wei; Liu, Renting, "***An ant-based distributed routing algorithm for ad-hoc networks***" Source: 2004 International Conference on Communications, Circuits and Systems, v 1, 2004, p 412-417.

[30] Ohtaki, Yoshitaka; Wakamiya, Naoki; Murata, Masayuki; Imase, Makoto, "***Scalable ant-based routing algorithm for ad-hoc networks***" Source: Proceedings of the Third IASTED International Conference on Communications, Internet, and Information Technology, 2004, p 50-55.

[31] Ahmed, Tarek H., "***Simulation of mobility and routing in ad hoc networks using ant colony algorithms***" International Conference on Information Technology: Coding and Computing, ITCC, v 2, Proc. ITCC 2005, p 698-703.

[32] Shen, Chien-Chung; Huang, Zhuochuan; Jaikaeo, Chaiporn, "***Ant-based distributed topology control algorithms for mobile ad hoc networks***" Source: Wireless Networks, v 11, n 3, May, 2005, p 299-317.

[33] UCLA Parallel Computing Laboratory and wireless Adaptive Mobility Laboratory, "***GloMoSim: A Scalable Simulation Environment for Wireless and Wired Network***".