Optimization of Mobile Agent's Communication Path in Complex Networks

Akomolafe Oladeji Patrick, Akanbi Awwal Olalekan

Department of Computer Science, University of Ibadan
Oyo, Nigeria

Abstract

Considering the pressing need to adopt more mobile agent functions to today's networks structure characterized by complexities, there is also a corresponding need for the optimization of the mobile agent movement and functions so that the aim of employing the use of mobile agent in these networks structure will not be defeated. Taking the need into consideration, this research work introduced a means for detecting a mobile agent's optimal path or route from a source to its destination where multiple paths or route exist by leveraging on the effect of link failure, time delay and mobile agent's propagation time.

Mobile agents have been used to manage simple networks efficiently in recent times but the evolution of complex networks have required a more systematic means of managing complex networks via mobile agents. As the network become larger and complex, Mobile agents become more vulnerable to network delay due to series of factors such as link overload, link failure, node failure. Different researchers have identified different means of optimizing the Mobile agent's path to solve this problem but latency, that is, time delay is still inherent due to unforeseen circumstances.

As the mobile agent starts from a known source node, available routes are first determined. As it proceeds from a particular node to the next node, time delay value is generated and the routing table is updated. This continues until the destination is reached.

Simulation results were obtained as measured for each route in the mobile agent network journey. The number of delays and time delay value for each delay journey between two nodes and the entire route propagation time were obtained and evaluated so as to determine which route had the least number of time delays and the least time delay value. This route was considered to be the optimal path in the mobile agent journey.

This research work provided an optimization scheme for mobile agents in complex networks by means of identifying the path with least time delay. With this research work proceedings properly implemented, there will is a drastic reduction in the rate of data loss by an agent in a network which is the utmost essence of employing a mobile agent in the network.

Keywords: *complex networks, link failure, propagation time, optimization of mobile agent movement, time delay*

---

1. Introduction

The complexity of real networks are growing in recent years. This has called for a commensurate upgrade in network management and performance so as to meet the demands of these growing complexities. Network Management is a major problem in the field of data communication. The major types of networks in question have dynamic complexity in nature.

A major fault in complex dynamic networks is the link and node failure phenomenon. Link and node failure most times occur as a result of fast movement of network nodes which in turn makes the network more unstable. The effect of this leads to network routing overheads which is not commonly found in static networks [1].

Deployment of Mobile agents is one of the most feasible methods of solving the complex network problems. It helps to improve the quality of service required for good network management by decentralizing and distributing network load for optimal performance (Makki and Wunnava, 2006).

In order to optimize the network for better performance in the view of reducing agent propagation time, time delay or even giving the chance for no delay which consequentially will help prevent packet loss, there is a need for an improvement on the existing routing methodologies. This paper made use of an adaptive ant colony algorithm customized to give allowance for the use of time delay parameter. This is because this paper gives more priority to time delay than the propagation time factor as time delay effect on packet loss is more critical to this paper work than time delay effect on propagation time which gives a shorter routing time.

2.Background

2.1 Networks

A network is a system that contains series of points or nodes, interconnected by communication links or path specifically for the aim of sharing information and resources. In networks, data is transferred in form of packets. Network components that originate, route and terminate the data are called network nodes. Data links are the components through which devices exchange data with each other. The process of transmission of data through data links in a network is called routing. The basic components of a network are shown in a network graph below.

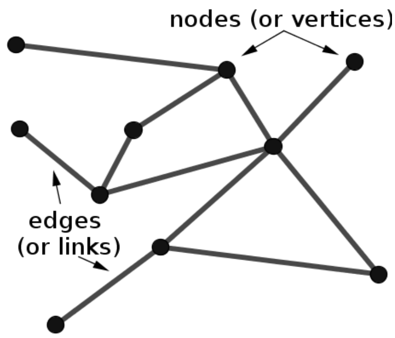


Figure 1: A typical Network Graph

2.2 Network Classification

Among many characteristic features to which networks are being classified are the degree of distribution, the clustering coefficient and the assortativity or disassortativity.

The simple, sparse random networks have a small clustering coefficient while the complex real world networks have a significantly large clustering coefficient. The clustering coefficient represents the density of triangles (connection of a node to others) within a network. This means that the complex real world networks have more triangle-like connections than the simple, sparse random network.

Degree Distribution of a Network

A network distribution degree can be defined as the number of connections between various nodes in the network, that is, the number of links that exist for a node with which it connects to other nodes in the network. Complex networks can exhibit complicated patterns with respect to the connections between the nodes in the network.

Clustering Coefficient

Representing a whole network in a graph, we will observe that the linking of nodes in the network graph form a triangular shape pattern. Some networks have theirs widely separated while others have their closely separated. A typical example of a network graph with clusters is shown below

2.3 Link failure in Networks

In finding a lasting solution to the performance of network operations these days, there is a need to review the cause of unhealthy message delivery on networks. With the advent of complex dynamic networks more adapted for operations, there had consequently been a tough quest to get this resolved easily as complex dynamic networks need more strategic handling than the simple networks mostly used in past technology era. Understudying the diverse reasons for packet failure and failure in message relaying, link failure have been considered as a major threat to optimal message delivery and optimal network performance in networks, especially in complex dynamic networks.

With the rate at which networks today are evolving, there is a more interplay of the occurrence of more complex dynamic systems. The emergence of the more complex dynamic systems has consequently led to the existence of more complex dynamic network. Link failure which causes more network routing overhead among other effects occur due to the tendency of the dynamic network to result to an unstable network. This is as a result of the rapt movement of the network nodes which dynamic networks exhibit [1].

When a link fails, it may cause routing update messages propagate to the rest of the network. There is a routing “cost” associated with such update propagation. Different parameters and views makes it clear that not all link failures are equal in terms of their routing “cost” [11]. Failures occurring at different locations in the network seem to result in different amount of update exchanges and different convergence time.

2.4 Mobile Agent

Early researchers such as Pattie Maes laid emphasis on agent’s autonomy in his own definition [22], Michael Coen concentrated his definition to focus on the negotiation and coordination of information transference by agents (M. H. Coen, SodaBot, 1994), while Stan Frankline and Art Graesser’s definition captures the essence of an agent [27].

In this research work piece, we define mobile agent as a computer program that can move autonomously from a source and continue its execution on the destination. Mobile agents are software agents with the notable characteristics of autonomy, social ability, learning, and most significantly, mobility. More specifically, a mobile agent can transport its state from one environment to another, with its data intact, and be capable of performing appropriately in the new environment. Mobile agents decide when and where to move. Just as a user directs an Internet browser to "visit" a website (the browser merely downloads a copy of the site or one version of it in the case of dynamic web sites), similarly, a mobile agent accomplishes a move through data duplication. When a mobile agent decides to move, it saves its own state, transports this saved state to the new host, and resumes execution from the saved state.

Mobile Agent Cycle

A mobile agent system is a platform that can create, interpret, execute, transfer and terminate mobile agents. An agent is designed to execute certain tasks. To start its journey, it is initiated from a source node, gets to the destination node and then returns to the source node to deliver the needed information.

Only mobile agents enter transit state, stationary agents do not. This ensures that a stationary agent executes all of its instructions on the node where it was invoked. The major agent actions are the move and execute action. The move action puts the agent in a transitory state while the execute action brings the agent out of the transitory state. The move action can only be initiated by the agent while the execute action can only be initiated by the agent system. To enter the transit state, the mobile agent initiates a mobility protocol that sends a move action to the agent management system. In the same accord, a mobile agent is brought out of transit state by an agent management system issuing an execute action upon the agent code [6].

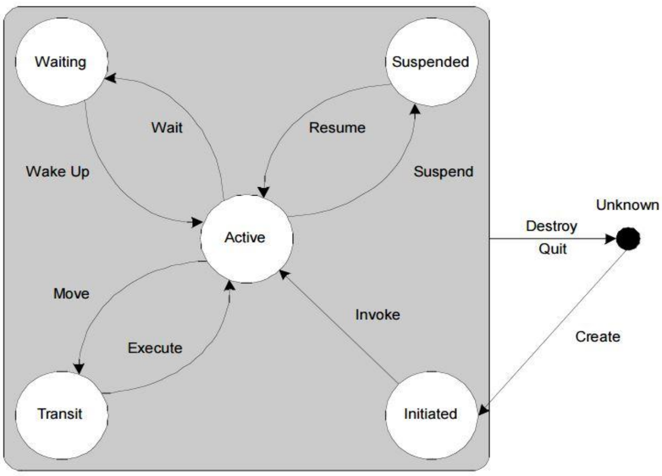


Figure 2: Mobile Agent Life Cycle [6]

Mobile Agent Network Management

With so much of network optimization problems in the past years, research direction has turned to the use of mobile agents to effectively manage networks optimally. This is being achieved by the aid of network load distribution which mobile agent do. This allow networks to conform to the needs of the fast evolving computing environments and helps to easily automate tasks efficiently. [15, 25].

The major purpose of a network management system is to provide basis for remote administration of network components and its hosts (nodes). A standard network management system has its basic structure containing several managed nodes which provides remote access to management tools.

2.5 Mobile Agent Routing Optimization in Networks

Combinatorial optimization gives a view of finding an optimal way of achieving a network task efficiently in cases of developing network issues. Some of the examples of this optimization process include network flow, shortest path problem, transport problem, location problem, routing problem, Critical Path Analysis and Program Evaluation & Review Technique, to mention a few.

Optimization is a phenomenon that gives a measure of the performance of computational algorithms in numerical software development. Performance optimization analysis has been demonstrated by the need to accommodate an increasing degree of efficiency, flexibility and adaptability. Optimization analysis for mobile agents introduces the study of message communication and computational algorithms with the aim to optimize traffic load and time management in mobile agent routing networks. Message communication affects communication performance while computational algorithms affect time performance [5].

Optimizing the routing technique for mobile agent involves determining the measures that best achieves the goal which the mobile agent is to achieve without using much of resources. In the course of the years during mobile agent researches, many algorithms have been developed and each one designed to work towards the aim of doing the job with less resources being used. Examples of these algorithms are the Ant Colony Optimization (ACO), Swarm Intelligence among others. Each of these algorithms have their strength and weaknesses, that is why there is a call to carefully observe and determine the functions to be adopted for a specific mobile agent task. Among these algorithms, the ACO is seen to be best for managing complex networks due to its nature of handling complex structures.

Ant Colony Optimization

Ants are small individual type of insect in the animal kingdom, probably one of the smallest. But when they perform operations collectively, they can form a very complex network structure, especially in their search for food. This happen due to the nature with which they search for their food. The interesting thing about the ant behaviour pattern is that they work hand-in-hand to achieve a common task which is one of the attributes that multi-agent systems leverage on. This network pattern which they form is referred to as colony. Example of an ant colony is shown in the figure below.

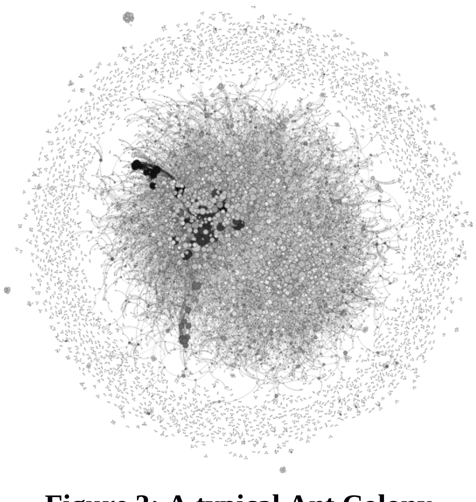


Figure 3: A typical Ant Colony

Biological study about ants show that they have a pheromone which is a fluid they secret and pass on as they move along any path. This is the interesting concept that the Ant Colony Optimization had built the ACO algorithm on. The pheromone as it is helps to guide the direction of movement. Mobile agent practice using the ACO algorithm is synonymous to the real life ant system as mobile agent can be as light weighted as the ants, move to execute a task from a source to a destination and back to the source and lots more attribute.

The Pheromone as it is evaporate over time, so there is a need for an update to be made on a path by the same ant or another ant adding more pheromone. The routing probability of real life ants is given by the width or density of pheromone on different paths. A path with the biggest or most dense pheromone concentration is the best path to follow as it literarily means that it is a path that has been mapped out by other ants also and that is where other are getting food. So therefore, an oncoming ant will most probably follow that path than to follow another and start looking for food. This concept is the basis to which the ACO algorithm uses to train a system to adapt and make precision at any time of need in a particular system.

In the search for a new source of food, ant randomly search for the best path. In this search, some pass through a long path while other through short path. But it is observed that because of the long time it takes for an ant to finish its journey on a long path, the ant chooses to follow the short path and consequently, other ants also follow suit by journeying through the short path, thus making the network of search form a colony.

The idea that application networks can use ant-like mobile agents to control communication networks was introduced in 1990 [26, 2]. The use of mobile agents routing protocols that exhibit the ant traits have been proposed for telecommunication networks because of the complex nature of the telecommunication networks and also because colony-like ants have a remarkable ability of solving complex problems in a distributed way [28, 3].

When a link fails, the agent will have to wait for the recovery of this link before the agent can proceed on its journey. As a result, some quality time had been wasted by the agent in delivering and also, it is made known that in waiting due to link failure, packet losses or information can be lost. The agent on a failed link is said to have breached its optimal performance due to the time delay factor. We regard both the lesser possibilities of link failure as well as a short time of movement although the time delay gains more priority than the propagation time which is because data loss is considered of more importance than the speed. This therefore might mean that a long route can be considered optimal if it has the least meaningful time delay. Notwithstanding, this does not erase the fact that we want the agent to travel both with little delay and little propagation time.

3. Optimization Procedure

Networks are identified by various metrics of which a major one is the degree of distribution of nodes and the numbers of links to a node. Knowing fully well that the complex dynamic network has numbers of interwoven nodes and links, we used an algorithm which caters to this effect. The algorithm used in this research work is a modified Ant Colony Optimization with the new form taking some alterations from the ant algorithm most especially to handle routing, mode of message delivery and optimization technique to achieve the aim of this research work.

3.1 Mobile Agent Route Optimization Procedure

The kind of network considered in this research work is a complex network. Knowing that we had a complex network, we had a heavier task of identifying the best route for a mobile agent than in a simple network having little or no interlocked links.

Majorly, there was the challenge of which node was the next to follow or which link of a node consisting of multiple links was the best to follow. Also entitled in this research was the issue of bad links or link failure in the complex network. Figure 9 below shows the directional procedure of how we achieved the purpose of path optimization.

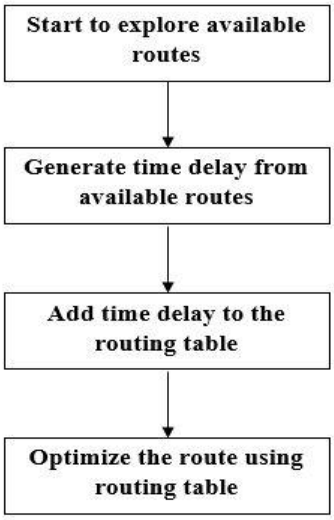


Figure 4: Mobile Agent Routing Implementation Procedure

Available Route Determination/Exploration

When moving from a node to another, there is a particular problem the agent encounter which is as regards to the choice of part it should take when there are multiple possible path it can take. These paths are composed of node-links. This is owing to the fact that a node in a complex network can have multiple links that are potential optimal paths for a mobile agent.

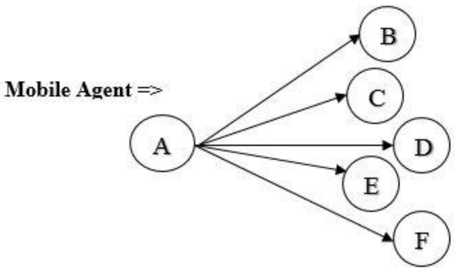


Figure 5: Typical example of a node A, with many node-links

From the figure above, many links originate from node A and encroaches node B, C, D, E, and F respectively. Where the problem lies is that the Mobile Agent moving from node A will have a problem in determining the link to follow into the next node as it journeys along a long path with many nodes.

To this effect, we made use of a random algorithm which helped to perform the operation of selecting a link to follow by the Mobile Agent.

Generating Time Delay

Having employed the use of the random algorithm to determine the next link which the mobile agent will migrate through in the course of fulfilling its mission, there was a very important condition that was considered for its success. Out of all the links of a single node, there's the probability of the existence of a link failure. A link failure typically appears as a period of consecutive packet loss that can last for many seconds, followed by a change in delay after the link is re-established. There will be a delay when there is a link failure. The time it takes the link to be recovered is termed to be the time delay for a mobile agent along a node-link.

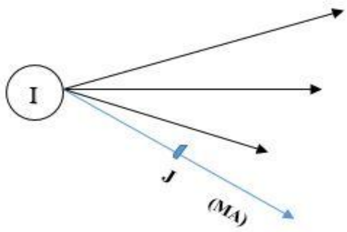
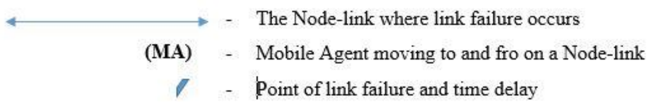


Figure 6: Network Node-link showing a link failure



Let the time it takes to get to Point J = Tjx
Let the time it takes to leave Point J = Tjz
Time delay = Tjz – Tjx

Addition of Time Delay to Routing Table

After the time delay had been generated from the section above, there was a need to update the routing table as the mobile agent moves along the path. The routing table is used to train the mobile agents so that subsequent movement will be quick to achieve, just like an ant leaves a pheromone on the path it has passed so that another ant can know the best path to move along. The table below shows the Routing table prototype.

Table 1: The Forward Movement Routing Table of a Mobile agent

S/No.	Initial Node	Final Node	Time Agent spent on link [secs]	Time Delay on link [secs]
1.	A	B	T_{ab}	Tab_d
2.	B	D	T_{bd}	Tbd_d
3.	D	E	T_{de}	Tde_d
4.	E	I	T_{ei}	Tei_d
5.	I	K	T_{ik}	Tik_d

From the table 1,
Source Node = A
Destination Node = K

Optimizing the Route using the Routing Table

From the routing table, we can vividly deduce the path the mobile agent followed to achieve its mission of successfully moving from the source node to the destination node. This operation was carried out for a to and fro movement of the mobile agent. Since the network class in question is a complex network, there were other paths on the network that can also enable the mobile agent successfully achieve its goal. So it was therefore imperative to explore the possibilities of other paths and the best path with the least link failures and least time was labeled as the optimal path on the network.

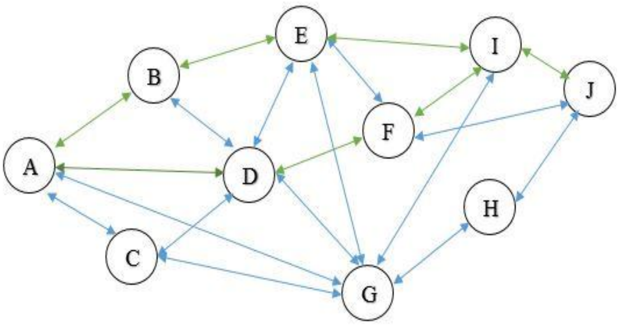


Figure 7: Complex Network showing Mobile Agent Successful Movement from Source to Destination

Source Node = A
Destination Node = J
Successful Agent Paths are:
i)A => B => E => I => J
ii) A => D => F => I => J

The Optim al path was selected from the above using the few conditions as it was applied in their ascending order. The conditions are:

i. Path with least link failures and time delay
ii. Link with least time spent from the source node to the destination.

Algorithm
Begin*;*
Determine source and destination nodes;
Determine number of source node links;
Repeat *until the source node is exhausted;*
Initiate forward movement;
Repeat *until agent reaches destination node*
Make path request;
Initiate movement and update routing table;
If *link failure then*
Update time delay;
Interchange destination with source and vice versa;
Initiate backward movement using the routing table
Repeat *until agent reaches destination node*
If *link failure then*
Update time delay;
End*;*

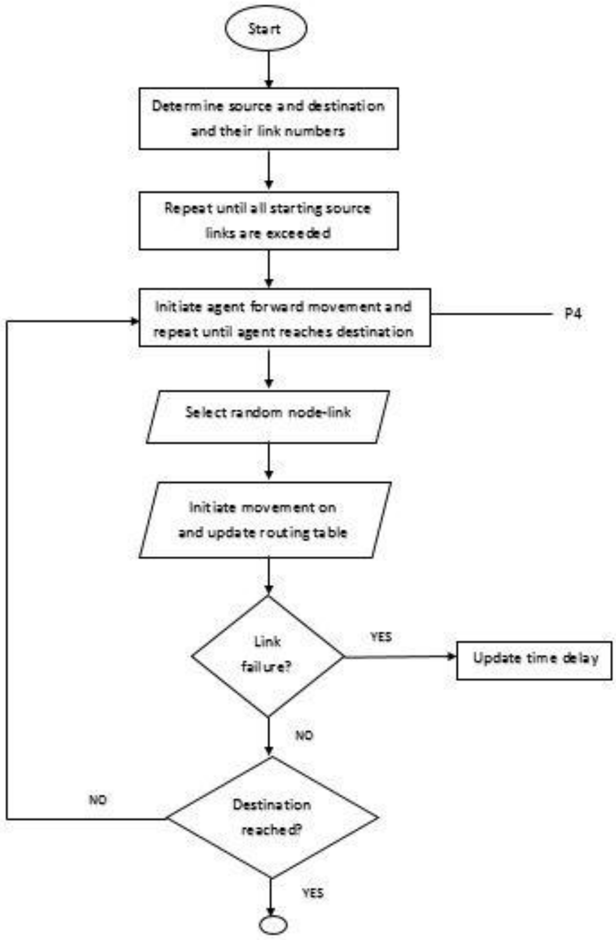

Start
Determine source and destination and their link numbers
Repeat until all starting source links are exceeded
Initiate agent forward movement and repeat until agent reaches destination
P4
Select random node-link
Initiate movement on and update routing table
Link failure?
YES
Update time delay
NO
Destination reached?
NO
YES

Figure 8:

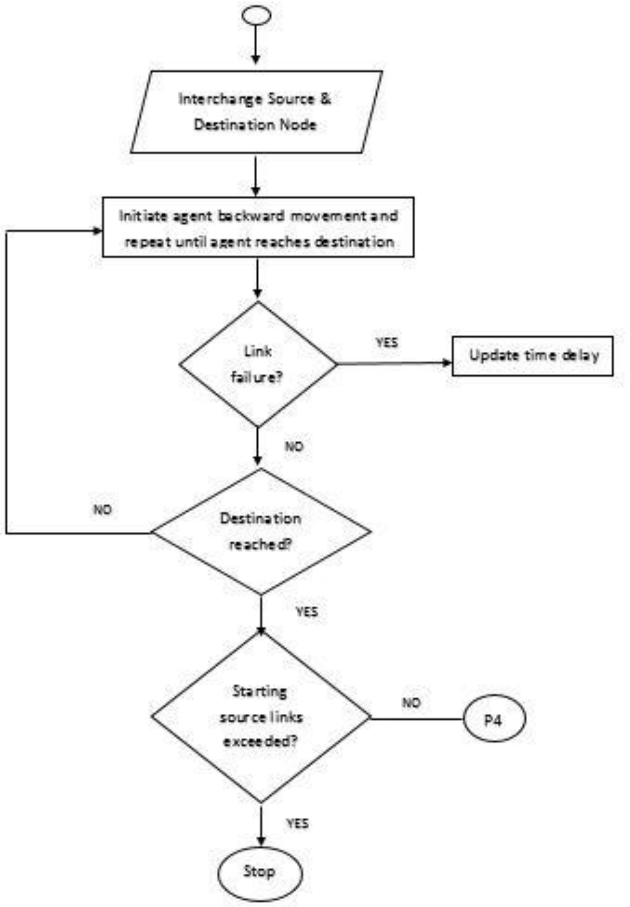


Figure 9: Flow Chart showing the Route Optimization Process

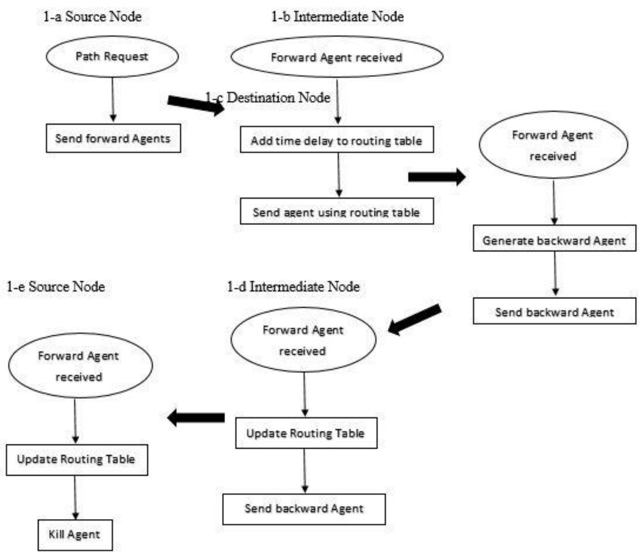


Figure 10: Optimal Path Determination Simulation

4. Result and Evaluation

4.1 Simulation Result

The table below shows the result of the simulation process of the complex network model. This result was read and analyzed from the output file generated after stopping the simulation process.

Table 2: Table showing the routing path from source to destination of the first link of the source node

S/No.	Initial Node	Final Node	Time Agent spent on link [secs]	Time delay (propagation delay) on link [secs]
1.	rte[1]	rte[4]	0.0204255	0.0
2.	rte[4]	rte[5]	0.0205337	0.0
3.	rte[5]	rte[10]	0.0206419	0.0
4.	rte[10]	rte[13]	0.0207501	0.0
5.	rte[13]	rte[18]	0.0208583	0.0
6.	rte[18]	rte[19]	0.0209665	0.0
7.	rte[19]	rte[26]	0.0210747	0.0
8.	rte[26]	rte[32]	0.0211829	0.0
9.	rte[32]	rte[34]	0.0583846	0.0204255
10.	rte[34]	rte[40]	0.0584928	0.0
11.	rte[40]	rte[52]	0.0586010	0.0
12.	rte[52]	rte[47]	0.0587092	0.0
13.	rte[47]	rte[48]	0.0588174	0.0001041
14.	rte[48]	rte[50]	0.0589256	0.0
		Total	**0.5183642**	

Table 3: Table showing the routing path from source to destination of the second link of the source node.

S/No.	Initial Node	Final Node	Time Agent spent on link [secs]	Time delay (propagation delay) on link [secs]
1.	rte[1]	rte[4]	0.0204249	0.0
2.	rte[4]	rte[5]	0.0205322	0.0
3.	rte[5]	rte[10]	0.0206419	0.0
4.	rte[10]	rte[13]	0.0207480	0.0
5.	rte[13]	rte[18]	0.0208565	0.0
6.	rte[18]	rte[19]	0.0209655	0.0
7.	rte[19]	rte[26]	0.0213740	0.0
8.	rte[26]	rte[32]	0.0311823	0.0105255
9.	rte[32]	rte[34]	0.0583842	0.0204255
10.	rte[34]	rte[40]	0.0584922	0.0
11.	rte[40]	rte[52]	0.0586002	0.0001041
12.	rte[52]	rte[47]	0.0587076	0.0
13.	rte[47]	rte[48]	0.0588170	0.0
14.	rte[48]	rte[50]	0.0589250	0.0
		Total	**0.5173451**	

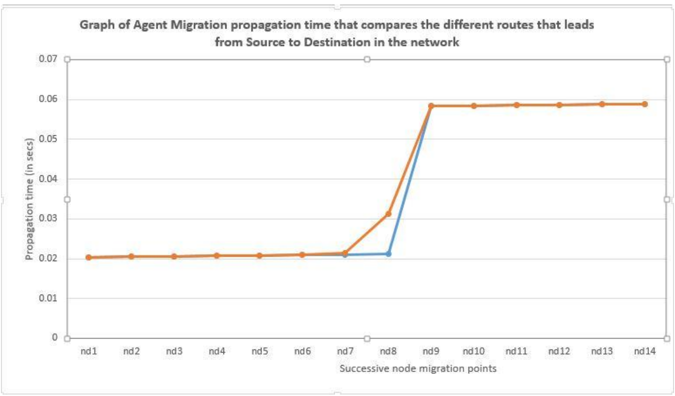


Figure 11: Propagation time for the Mobile agent routing along different routes

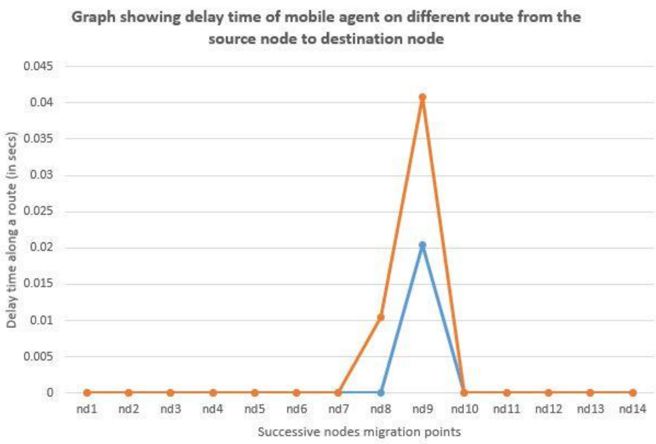


Figure 12: Time delay by the Mobile agent during routing along different routes

From Table 4.1,
Total travel time for the agent = 1.0367284
Total delay = 1
Total delay time = 0.0205296

From Table 4.2,
Total travel time for the agent = 1.0346902
Total delay = 2
Total delay time = 0.0310551

From comparison between the two table values, using the measure metric stated in chapter 3, the path/route represented in Table 4.1 is the optimal path because it has the least link failure and time delay. This is because with little time delay and link failures, the assurance of safe delivery of data transported by the agent is made to effect.

Therefore, the optimal path/route is
rte[1] – rte[4] – rte[5] – rte[10] – rte[13] – rte[18] – rte[19] – rte[26] – rte[32] – rte[34] – rte[40] – rte[52] – rte[47] – rte[48] – rte[50].

From the simulation result and the graph, the available routes for the mobile agent's journey were the routes in which the mobile agent moved from the source to the destination and back to the source node without total failure. The available links of a node were determined using the random algorithm.

In addition, among all the available routes in the network that leads from the source to the destination and back to the source, one was considered as the most optimal. The links not selected that leads out from a node are unavailable links this optimal route was determined using various time delay and propagation time parameters with the time delay gaining more attention. The optimal path was identified from the table information due to the calculated time of propagation and time delay as made know in the sections above.

5. Conclusion

This research work presented an optimization technique for mobile agent routing in complex and real-life networks based on time delay and network routing time parameters. In the complex route containing many links that links the source node to other nodes, a random algorithm was used to select the available route. The available link picked at an instance was the randomly selected link which leads out to another node in the mobile agent journey.

In the mobile agent journey, the links not selected that leads out from a node are unavailable links. Links were selected and a time delay was checked on each link. This time delay was used to update the routing table. Some parameters and conditions were used to evaluate the optimal path for the mobile agent in the order of time delay and the total propagation time spent on the Journey. The time delay was given more priority because it causes packet loss and lesser data will be lost on a short time delay rather than the total propagation time. For evaluation, the different path that links the source to destination nodes were compared based on the major model parameters which are the time delay and the total completion time.

Recommendation

This work has laid down a concrete means of optimizing a mobile agent routing system. The research work actively worked on the time delay perspective in determining the optimal path out of multiple numbers of paths that link the source to the destination. It is however clear that time delay is a big problem in the execution of mobile agents adapted to complex system networks most especially. Link failure causes time delay, which was used as a measuring parameter, and packet loss or mobile agent data loss. This effect of link failure which is perceived to cause agents to loose data is a big problem which this solution can help provide answers to.

It is therefore recommended that in subsequent extension of the work, the research should duly deal with the issue of both time delay and link cost simultaneously.

References

[1]. Al-Rabayah, M., & Malaney, R. (2012). A New Scalable Hybrid Routing Protocol for VANETs. *Vehicular Technology, IEEE Transactions on*,61, 2625-2635.

[2]. Appleby, S., & Steward, S. (2000). Mobile software Agents for control in telecommunications Networks. *Br. Telecom Technol. J.*, 18, 68-70.

[3]. Bonabeau, E., Henaux, F., Guérin, S., Snyers, D., Kuntz, P., & Theraulaz, G. (1998,) *Routing in Telecommunications Networks with Ant-Like Agents. In Proceedings of the Second International Workshop on Intelligent Agents for Telecommunication Applications.* Paris, France : Springer Berlin Heidelberg .

[4]. Cauvery, N. K., & Vis Wanatha, K. V. (2008). Enhanced Ant colony based algorithm for routing in Mobile Ad hoc Network. *World Acad. Sci. Eng. Technol*., 22, 30-35.

[5].]Faiz, S., Hesham, A., Khalil, S., & Mahmoud, O. (2012). Evaluating Time Performance Optimization Analysis for Mobile Agent Message Communication Using Assignment Computing Agent. *Journal of Applied Sciences*, 12, 1290-1296.

[6]. Foundation for Intelligent Physical Agents. (2000). *FIPA Agent Management Specification*. Retrieved from http://www.fipa.org/specs/fipa00023/SC00023K.html

[7]. Fruchterman, T. M. J., & Reingold, E. M. (1991). *Graph Drawing by Force-directed Placement.* Retrieved from https://web.stanford.edu/~messing/CreateBasicNetVis.html

[8]. Genco, A. (2008). Mobile Agents: Principles of Operation and Applications. *B and 6 von Advances in management information series*.

[9]. Gianluca, I., Chen-Nee, C., Richard, M., Supratik, B., & Christophe, D. (2002). *Proceedings of the second ACM SIGCOMM Workshop on Internet measurement* - IMW '02, New York, NY, USA : Association for Computing Machinery (ACM).

[10]. Grandjean, M. (2014). La connaissance est un réseau. Les Cahiers du Numérique.*Digital Papers*, 10, 3, 37-54.

[11]. Jonatan, S., & Elias, P. D. (2007). Fault-Tolerant Dynamic Routing Based on Maximum Flow Evaluation. *Dependable Computing* Volume of the series *Lecture Notes in Computer Science*.Springer Berlin Heidelberg.

[12]. José, L. O., & Rui, P. L. (1999). *Distributed Management Based On Mobile Agents*, 2-13.

[13]. Julia, P. C. (2012). *Evolutionary Games in Complex Topologies.Interplay between Structure and Dynamics.* Berlin Heidelberg : Springer-Verlag.

[14]. Linna, F., & Jun, L. (2010). A free-roaming Mobile Agent security protocol against Colluded truncation attack. *2nd International Conference on Education Technology and Computer (ICETC)*, 5, 261-265.

[15]. Liotta, A., Pavlou, G., & Knight, G. (2002). Exploiting Agent mobility for large-scale Network monitoring. *IEEE Networks*, 16, 7-15.

[16]. Lange, D. B., & Oshima, M. (1999). Seven Good Reasons for Mobile Agents. *Communication of the ACM*, 42, 88-89.

[17]. Mazda, A. M., Arnak, V. P., Ashot, N. H., & Naira, M. G. (2013). *An Anomaly Event Correlation Engine: Identifying Root Causes, Bottlenecks, and Black Swans in IT Environments*. Retrieved from https://labs.vmware.com/vmtj/an-anomaly-event-correlation-engine-identifying-root-causes-bottlenecks-and-black-swans-in-it-environments

[18]. Newman, M. E. J. (2004). The structure and function of complex networks. *Department of Physics, University of Michigan, Ann Arbor, and Santa Fe Institute*, 45, 167-256.

[19]. Nykamp, D. Q. (n.d.). *One of the simplest types of Networks*. Retrieved from http://mathinsight.org/simplest_type_of_network

[20]. Nykamp, D. Q. (n.d.). *One of the simplest types of Networks*. Retrieved from https://www.sg.ethz.ch/media/publication_images/RnD-Net.png

[21]. Nykamp, DQ. (n.d.). *One of the simplest types of Networks*. Retrieved from http://mathinsight.org/network_introduction

[22]. Pattie, M. (1997). On Software Agents: Humanizing the Global Computer. *IEEE Internet Computing*, 1, 10-19.

[23]. Restrepo, J. G, Ott, E., & Hunt B. R. (2007). Approximating the largest eigenvalue of Network Adjacency matrices. *Phys Rev E*, 76.

[24]. Roth, M., & Wicker, S. (2004). Asymptotic Pheromone Behavior in Swarm Intelligent MANETs. *In Proceedings of the Conference on Mobile and Wireless Communication Networks*, *162*, 335-346.

[25]. Rubinstein, M. G., Duarte, O., Pujolle, & G. (2002). Scalability of a Network management application based on Mobile Agents. *J. Communications and Networks*, 5 ,240-248.

[26]. Schoonderwoed, R., Bruten, J. (1997). Ant-Like Agents for Local Balancing in Telecommunications Networks. *In Proceedings of the First International Conference on Autonomous Agents*, 209–216.

[27]. Stan, F., & Art, G. (1996). Is It an Agent, or Just a Program? A Taxonomy for Autonomous Agents. *Proceedings of the Third International Workshop on Agent Theories, Architectures, and Languages*, 1193, 21-35.

[28]. Subramanian, D., Druschel, P., & Chen, J. (1997). Ants and Reinforcement Learning: A Case Study in Routing in Dynamic Networks. *In Proceedings of the Fifteenth International Joint Conference on Artificial Intelligence*, 832–839.

[29]. Watts, D. J & Strogatz S. H (1998). Collective dynamics of small-world Networks. *Nature,* 393, 440-442.

[30]. Zhong Z. (2011). Ant Colony Algorithm Based on Path Planning for Mobile Agent Migration. *Power Electronics and Engineering Application, 23, 1-8.*