INVESTIGATION OF ANT COLONY ALGORITHM IN MULTIPLE TRAFFIC FLOW ENVIRONMENTS

Ali Tizghadam
University of Toronto
atizghadam@comm.utoronto.ca

Massoud Hashemi
Isfahan Univ. of Tech.
hashemim@cc.iut.ac.ir

Alberto Leon-Garcia
University of Toronto
alberto.leongarcia@utoronto.ca

Abstract

The conventional approaches to routing and bandwidth allocation, the two major components of traffic engineering, have proved insufficient to address QoS requirements of flows while optimizing utilization for complex communication networks. In this paper we consider ant colony algorithms to address this problem. Our studies show that the ant-based routing models are sensitive to initial parameters settings. Only careful adjustments of these initial parameters results in an acceptable convergence behavior. The robust behavior of the real ant compared to the routing algorithms derived from it inspires us to investigate the reasons behind the shortcomings of these algorithms. We present results from an in-depth study of ant behavior in a quest for a robust algorithm. In this work we consider a realistic environment in which multiple source-destination flows compete for resources. We study the routing and load balancing behavior that emerges and show how the behavior relates to analytical approaches. We show the results using simulations in OPNET and derive recommendations on the improvement of the ant-like algorithms.

Keywords: *Traffic engineering; ant algorithm.*

1. Introduction

Traffic engineering is one of the active research areas in communication networks. The traditional form of routing and resource allocation, as the two major building blocks of traffic engineering cannot address quality of service requirements of flows while optimizing network utilization for complex communication networks. In this paper we consider ant colony algorithms to address this problem. In this approach foraging ants find the shortest path in a synergistic way. While moving back and forth between nest and food, ants mark their paths by secreting pheromone. Step-by-step routing decisions are biased based on the local intensity of pheromone field which is the colony's collective and distributed memory. Ants will follow the most dense route in a maximum likelihood way. The actual algorithm implemented in nature by real ants is slow in convergence.

Our studies show that the ant-based routing models are sensitive to initial parameters settings. Only careful adjustments of these initial parameters results in an acceptable convergence behavior. The robust behavior of the real ant compared to the routing algorithms derived from it justifies the investigation of these algorithms in depth to find the reasons behind their shortcomings. We present results from our study of ant behavior in a quest for a robust algorithm. Most of the ant-based algorithms have been studied with limited source-destination traffic. In this work we have extended the algorithm to a more realistic environment in which multiple source-destination flows compete for the resources. We study the routing and load balancing behavior that emerges and show how the behavior relates to analytical approaches for optimal minimum delay algorithms by Gallager [3], Mitra [6], and others [4], [5]. We show the results using simulations in OPNET and derive recommendations on the improvement of the ant-like algorithms to achieve load balancing.

The rest of this paper is organized as follows. In chapter 2 we highlight the problems of traffic engineering. Chapter 3 is dedicated to the ant algorithm to provide the overall view of the ant approach. In chapter 4 we briefly describe the outstanding analytical methods introduced to address the problem of dynamic routing and flow assignment. Our experiments and the results including our view of the ant algorithm is discussed in chapter 5. Finally we conclude the paper in chapter 6.

2. Traffic Engineering

Traffic engineering is the process of mapping traffic flows onto the physical topology to meet traffic requirements, to enhance overall network utilization and create a uniform distribution of traffic throughout the network.

Traffic engineering in the traditional Internet is achieved by manipulating routing metrics, such as monetary cost, hop-count, bandwidth, reliability and delay. Since IGP (Interior Gateway Protocol) route calculation is topology driven and based on a simple additive metric such as the hop-count, it does not consider other important dynamic criteria such as bandwidth availability. As a result, traffic can be unevenly distributed across the network causing inefficient use of resources. Uneven distribution of traffic is complicated since it can be the product of the dynamic routing protocols such as OSPF and IS-IS, that select the shortest paths to forward packets. Hence a solution is required that takes into account more factors than the common path-metrics. While using shortest path conserves network resources, it may cause some other problems, such as congestion on some paths and under utilization of other paths.

0-7803-8886-0/05/$20.00 ©2005 IEEE
CCECE/CCGEI, Saskatoon, May 2005

These known problems with routing and the trends in networking and telecommunication provide incentive to look for another approach in routing and flow assignment as two main parts of traffic engineering.

3. Ant Algorithm

3.1. History

Nature has always been an important source of inspiration for academic research. In particular there is much interest in the behavior of ants. Individual ants seem to move at random, do nothing but wander off, and yet groups of ants can accomplish complex tasks. Somehow, a collective intelligence is formed out of many simple elements which is called **swarm intelligence**. Each agent (ant) processes a very simple algorithm. The collective outcome realizes a much more complex algorithm. The whole system is distributed and adaptive. Ants cannot see or hear. They only sense the environment, and also the food. Ants cannot talk either, they communicate indirectly through the environment. An ant can leave a trial of pheromones which are materials with particular fragrance. Ants can smell and sense the pheromones left by other ants, moreover, ants can detect the density of the pheromones.

3.2. Finding the Shortest Path

Ants find the shortest path to food according to this procedure: two ants start their random walk. They both eventually find the food. The one taking the shorter path finds the food first. Each ant leaves a trail of pheromones behind. Having taken the food the ants follow their pheromone trail towards the nest. The one with the shorter path returns first and arrives back to the nest first. Now a third ant wants to search for food. The ant realizes the trials left behind by its predecessors. **Most likely** it follows one of the existing trials rather than initiating a new trial and **most likely** it follows the trial with the **higher density** of pheromones. This results in **even denser** pheromone trial on the shorter path and in the long run this results in most ants using the shortest path.

When an ant starts its walk with some small probability it starts a new trial. The first ants may not necessarily have chosen the shortest path but starting new paths helps continue the quest for shorter paths until finding the shortest one and eventually the ants emerge around the shortest path. The pheromones **evaporate** over time. This is an essential requirement for the **dynamism** of the algorithm. The algorithm is adaptive because of the evaporation and the fact that ants keep starting new paths with some probability.

3.3. Ants in Communications Networks

Ants in networks are emulated by mobile agents. Mobile agents are carried by packets. Special packets can be used as mobile agents (ants). Pheromones pass the information about the length of the path (time) to other ants. The agents can pass the same information to data packets at the nodes. Ants decide based on the density of the pheromones and some probability values. The probability values can be calculated based on the path information and listed in routing tables in the nodes. Starting with a static routing table for each node, every individual routing table stores the probabilities of using the next hops to reach all possible destinations. The sum of all the probabilities at each row should be equal to one.

Different methods have been used in the literature for implementing and updating the routing tables using the ant approach such as AntNet [1].

4. Analytical Approaches

In this section we will briefly review existing analytical approaches to addressing the traffic engineering issue. Analytical routing algorithms can be distributed or centralized and also static or dynamic. Static routing algorithms cannot keep themselves up-to-date with the continuing changes in networks. Centralized methods suffer from the lack of scalability and having a single point of failure. Other distributed and dynamic routing and flow assignment algorithms have stability and convergence issues.

A common characteristic of these methods is their dependence on one or more heuristic parameters that are found based on experiments. In Gallager's algorithm [3] in order to avoid loops the algorithm uses a parameter η that should be globally chosen and every router must use it to ensure appropriate behavior but this parameter depends on input traffic pattern. It is impossible to find one working value for all input traffics.

In Mitra's approach [6] also a heuristic parameter is used which is critical for the robustness of the algorithm. This factor is called "bandwidth protection" R. In [5] again a heuristic is used which is in the form of a function.

5. Our View

In this section we present our study of the ant colony algorithm based on simulation and discuss the results. We followed [1] for implementing ant colony algorithm, but we also examined modifications to show how the ant approach can be exploited to achieve load balancing.

5.1. Methodology

Our simulations of the ant algorithm are applied to a fish-like network configuration as illustrated in figure 1. The network consists of four routers and four hosts at the edges. Ants are generated at routers regularly and addressed to the destination hosts randomly. We use a uniform distribution to assign the destinations to the ants which are called forward ants at this stage.

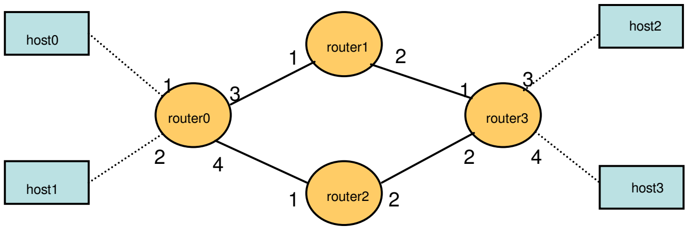

Figure 1

The forward ants are routed to output links at each router until they find their way to their assigned destination hosts. A trace of the route traversed by the ant is also stored in the ant. At the host the ants are transformed to backward ants and sent back to the source through the same path that they took to arrive to the destination. At each router across the path the ants will then update the routing table that is used to route data packets. In the original ant colony algorithm the same table is used to route forward ants as well.

Data packets are generated by the hosts and addressed to certain destination hosts to create data traffic flows. At each router data packets will be routed to output links based on the information listed in the routing table. In an ant colony algorithm the routing table contains a probability number for every destination host though every existing output link. Packets are directed to an output link in proportion to the link probability.

The forward ants measure the travel time from each interim router to the final destination. The travel time which is a reflection of the route conditions is used to update the probability table when the backward ants come back to the router.

Except for the routing table, each node also keeps a table with records of the **mean** and **variance** of the trip time to every destination (delay). At each node, backward ants update the trip time statistics to the destination in addition to the output link probability. We derived these equations for updating the probabilities [1]:

$$P'_{ij} = \begin{cases} (1-r)\times p_{ij} \ldots\ldots\ldots\ldots if \ldots\ldots\ldots j \neq k \\ (1-r)\times p_{ij} + r \ldots\ldots if \ldots\ldots\ldots j = k \end{cases} \quad \text{(Intermediate probability)}$$

This equation gives an interim probability value for destination i from this router. In this equation *j* refers to all output links and *k* is the link that backward ant came from.

The probability values are filtered according to the following equation before being set into the routing table. This is to reduce the variations in the table because of temporary increases in the travel times.

$$p''_{ij} = \alpha \times p'_{ij} + (1-\alpha) \times p_{ij}$$

r can be derived in different ways. We use the following to calculate r:

$$r = \beta \times \frac{1}{p_{00}} \times \frac{|d-m|}{\sigma}$$

In these equations α and β are two parameters that could be tuned, d, m, σ are the instant delay or trip time, mean of delay and standard deviation of the delay.

5.2. Simulations

We examined three different approaches. In the first approach we used the original ant colony algorithm. In this approach forward ants are routed at the routers using the probability values in the routing tables similar to data packets.

In the second approach we used round-robin to forward the forward ants to the existing output links. The probability values are used to route only data packets.

In the third approach we modified the probability calculation algorithm while using the same round-robin approach for forward ants.

5.3. Analysis and Discussion

First we consider the results from the first set of experiments based on the first approach. In the majority of scenarios using various settings of the parameters, the ant algorithm reveals a strong tendency towards finding and using a major route to the destination (which is most likely the fastest one). Other routes exist but with much less share of the carried traffic.

As shown in figure 2, even though links 3 and 4 of router 0 are exactly alike, the probability value for link 3 rapidly rises and as a result the traffic from source 1 to destination 2 eventually flows entirely through link 3.

This result seems to be in agreement with the actual behavior of ants in the real world which is basically directed in finding the best path to the food source while exploring other paths. This behavior leads to finding a robust path to the destination with enough dynamism to adapt to changes and to find new and better paths.

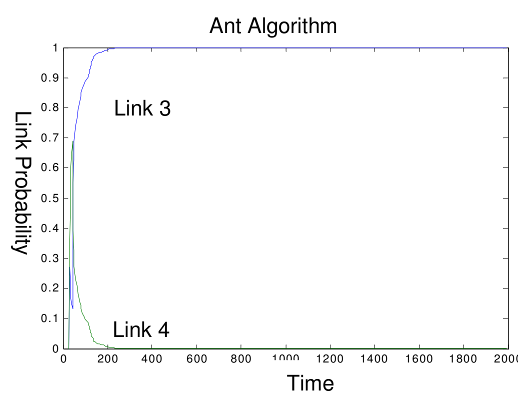

Figure 2

In a network of multiple sources and destinations with traffic flows from various sources to various destinations, the desired outcome involves distribution of data traffic across multiple routes from source to destination in a balanced way such that overall delay and utilization of the links is optimized over the network.

The ant colony implementation of ant behavior does not seem to be the best solution to this situation according to our observations of the results from the first approach.

In analytical solutions for the multi-commodity flow assignment problem, information about the links is used at the edge routers in a distributed way to calculate the best distribution of traffic into possible routes to achieve load balancing aimed at reducing delay and increasing link utilization.

Using probability tables for forwarding ants in the ant colony algorithm creates a feedback loop in favor of links that have better trip time results. This is the reason behind the convergence behavior of this method.

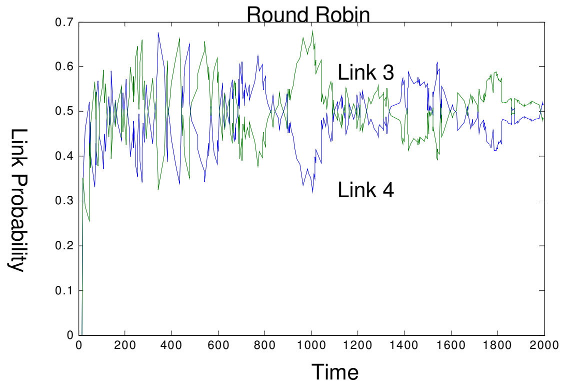

Figure 3

In the second approach we eliminate this mechanism and replace it with a round-robin selection of output links for forward ants. As expected the result as shown in figure 3, shows a better load balancing between existing possible links to the destination represented in the form of closer probability values.

Inspired by the analytical solutions, in a third approach we used local processing of the link information gathered by the ants to achieve a better load balancing, which also results in a better delay performance and link utilization. Our results, as shown in figure 4, show that a better and more robust load balancing (closer and more stable probability values) is achieved between link 3 and 4, compared to the previous cases.

Our experiments revealed that somewhere a friendly relationship between analytical and bio-inspired approaches has to be made. We believe the real ant in nature has a robust behavior but this process is slow and also is not necessarily promoting load balancing. While the evaporation property of ant pheromone brings some relevance to the concept of load-sharing, in fact the evaporation in ants is to provide the dynamism necessary for adaptive behavior. In other words ants have completely robust, adaptive and distributed behavior but this behavior is not aimed at load-balancing as an objective.

On the other hand there are some analytical solutions to balance the load in the whole network and to do resource allocation and resource assignment as a direct result.

Our findings show that there is opportunity for thinking differently. **We can be inspired by the wonders of the nature but we need not imitate them.** We are free to modify the bio-inspired approaches (ant in our discussion) to obtain new desired behavior.

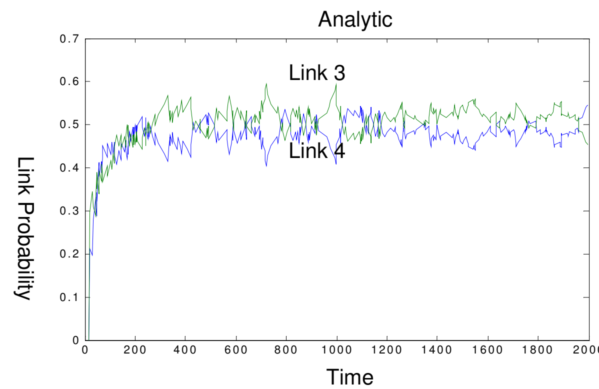

Figure 4

6. Conclusions and Future Works

In conclusion, our suggestion regarding the use of ant algorithms is that improved behavior is possible by augmenting it with the analytical computation.

Our current work involves applying analytical solutions to ant like algorithms for routing and flow assignment to extract a robust and autonomic routing algorithm capable of tolerating predictable changes in traffic patterns and learn the unpredicted ones.

References

[1] G.D. Cario, M. Dorigo, “AntNet: Distributed stigmergetic control of communication networks,” *Journal of Artificial Intelligence Research,* vol. 9, pp. 317-365, 1998

[2] Z. Subing, L. Zemin, “A QoS routing algorithm based on ant algorithm,” *International Conference on Communications (ICC),* vol. 5, pp. 11-14, June, 2001

[3] R.G. Gallager, “A minimum delay routing algorithm using distributed computation,” *IEEE Trans. On Communications,* vol. 25, pp. 73-84, January, 1977

[4] D. Bersekas, R.G. Gallager, “Second derivative algorithm for minimum delay distributed routing in networks,” *IEEE Trans. On Communications,* vol. 32, pp. 911-919, 1984

[5] J.J. Garcia-Luna-Aceves, S. Vutukury, W.T. Zaumen, “A practical approach to minimizing delays in internet routing,” *International Conference on Communications (ICC),* vol. 1, pp. 6-10, June, 1999

[6] E. Bouillet, D. Mitra, “The Structure and Management of SLAs in Networks", *IEEE JSAC*, vol. 20, no. 4 pp. 691-699 *(ICC),* vol. 1, pp. 6-10, June, 1999