An Ant Based Rate Allocation Algorithm for Media Streaming in Peer to Peer Networks

Amir Hesam Salavati, Hadi Goudarzi, Mohammad Reza Pakravan
School of Electrical Engineering
Sharif University of Technology
Tehran, Iran
Email: (salavati-h_goudarzi)@ee.sharif.edu, pakravan@sharif.edu

Abstract—In this paper, we propose a novel algorithm for rate allocation in multiple-source media streaming peer to peer networks. Our algorithm is based on ant-colony optimization and capable of handling network changes which occur quite often in unstructured P2P networks. The suggested algorithm does not need any information about the topology of the network. Moreover, it could get over uncertainties in network state information, particularly the rate of media provider nodes that could happen due to lack of accurate measurements. We show that our algorithm will reach the maximum achievable rate of the network quite fast and with relatively little overhead.

In our simulations, we have demonstrated that in cases where network state information is inaccurate, the suggested ant-based rate allocation method will lead to the same results that other optimization-based rate allocation algorithms yield. Moreover, we have shown that the proposed algorithm has an intrinsic low pass filter which discriminate between transient network changes from permanent ones. If the changes in the network is transient, the algorithm compensate the temporary losses quite fast and without much effort. In cases where the network changes last longer, the algorithm overcomes losses by employing other nodes that have the media stream available. The rate of adaptation is adjustable and must be carefully determined according to network conditions.

I. Introduction

Peer to peer networks have grown rapidly in recent years. Much of the Internet traffic is related to these networks nowadays. Although P2P networks were originally designed for file sharing purposes, they have evolved to provide streaming and live media contents as well. This is mainly due to the considerable increase in the bandwidth available to ordinary network users

Multimedia streaming has become one of the most important applications in the Internet. However, sending and receiving media streams causes some challenges for users due to large bandwidth consumption and real time nature of streams which forces users to download the media file first and then play it some later time.

Because of special characteristics of P2P networks, multimedia streaming in peer to peer networks has its own challenges. Peer nodes in a P2P network could join and leave the network at any time. Moreover, rate of peer links vary in time to a great extent. Since required bandwidth for receiving media streams is quite high and the output bandwidth of users are limited, a single user usually can not supply the necessary

streaming rate for other peers. Therefore, in practice multiple peers should provide media streams to another peer. The sum rate of these peers must be higher than the required rate of the media stream.

These two properties of peer to peer networks, namely network dynamism and the need for participation of several media sources, are the most important challenges of P2P media streaming. Any algorithm designed for peer to peer media streaming must be able to cope with these challenges.

Ant-based algorithms are based on a relatively new concept which is inspired by the collective foraging behavior of some ant species. Ants use a chemical substance called pheromone for indirect communications. There are several algorithms that model and exploit this behavior for solving different problems such as graph-based NP hard combinatorial optimization problems, e.g. the traveling salesman problem (for a quick review see [13]).

The foraging behavior of ants and file sharing in peer to peer networks are very similar. In the first case, there is an ant colony where ants start their journey from and bring food back to. Initially, ants do not know the location of food sources initially. Their task is to search for food and find those sources that are easier to access and then bring back the food to the colony. Since food resources are limited, it is best to find multiple food sources and supply the colony with all these sources.

Similarly, in peer to peer networks, there is a node requesting for a file or media stream. Since the node does not know which nodes could provide it with its required file or stream, it has to do a search across the P2P network, find those nodes that have the requested content available and start receiving data from these sources. Noting the lack of bandwidth and to increase reliability and distribution of traffic load in the network, it is best to receive the desired content from multiple sources. These similarities between the self-organizing behavior of ant colonies and self-organization in peer-to-peer networks have already been observed in [14]. Since they do not require any global knowledge about the network, ant algorithms are qualified for peer-to-peer networks. For a nice animated overview of ants foraging behavior see [15].

To even increase the similarities between peer to peer networks and ant colonies, we use Multiple Description Codes

(MDC). Internet was initially designed for best effort type of traffic. But in multimedia streaming, we have to deal with real time traffic. In real world, users are heterogenous in their access bandwidth to internet. Moreover, in multimedia peer to peer networks, there are multiple sources that provide the stream. As a result, we must use some sort of coding that breaks media stream into multiple descriptions, each of which could be received from different sources and must be decoded separately from other descriptions. The more descriptions a user receives, the higher quality its stream will be. Since unstructured peer to peer networks are highly dynamic, MDC provides a great opportunity to share multimedia contents across these networks.

In this paper we propose an algorithm for choosing serving nodes and do media rate allocation for these nodes based on foraging behavior of ants. We assume that media streams are coded with MDCs and each candidate source nodes contain multiple descriptions of the desired media stream. While the suggested algorithm is very simple, it is capable of reaching maximum achievable rate of media streaming in a P2P network. Our algorithm does not need any information about network topology and can be executed in a distributed manner. In addition, even if for any reason the rate of serving nodes is not their actual streaming rate, our algorithm works without any problem.

The rest of the paper is organized as follows: In section II we briefly review current works on media streaming rate allocation. Section III introduces ant-based routing algorithms. In section IV we give a detailed explanation of our ant-based rate allocation algorithm. Section V shows simulation results. Finally section VI concludes the paper and explains future works.

II. Related Works

Peer to peer media streaming has been a hot research area in recent years. Many researchers have worked on different aspects of media streaming in P2P networks. One of the most important works on multimedia streaming in overlay networks is oStream [1]. oStream investigates different streaming methods in one multicast session application layer networks. In oStream, it is assumed that one peer could provide necessary data rate for at least one other user so that media streaming could be implemented as a multicast communication.

Another paper that has examined P2P media streaming more deeply is [2]. ZIGZAG is an algorithm that builds multicast trees in peer to peer networks. Due to great network dynamism of peer to peer networks, ZIGZAG entails multiple mechanisms to monitor and maintain multicast trees.

While previous papers considered only the single source case, where one node could provide the necessary rate for streaming media contents to other peers, multi source streaming is discussed in some works like [3] and [4]. In [3], the authors presented a multi source streaming system where multiple sources transmit packets of multimedia content to the receiving node with some redundancy. This increases the reliability of media streaming in a network where packets can get lost and nodes may or may not be available during the desired streaming session. One of the important works that considers multiple source media streaming is [5]. It suggests a method to find the best set of nodes to provide media stream from all nodes that have a media content available. In this method, it is assumed that the rate of peer nodes are accurately measured and the topology is known by the users. Nodes run an optimization algorithm to maximize the expected rate of media stream.

In addition, a simple method is proposed to do rate allocation and monitor the streaming rate of peer users in the network called PROMISE [6]. PROMISE tries to find the optimum set and allocate proper rate when there are more than one intermediate routers between the peers. One of the disadvantage of PROMISE is the large amount of overhead caused by the optimization algorithm. IPROMISE is proposed [7] to solve the first issue. In [8] an algorithm is proposed to solve the second problem by implementing a suboptimal algorithm which has less overhead. This suboptimal algorithm first sorts peer nodes based on their availability. Then it adds each node to the active source list based on network topology. It is shown that while this method is suboptimal compared to PROMISE, its final rate is very close to what PROMISE's optimal algorithm yields. A new approach to choose the best set adaptively is proposed in [9]. In [10] authors have investigated methods of changing the quality of media stream for different users according to their network resources. To achieve this goal, authors have examined quality adaptation and layered media coding methods.

Due the heterogeneity of users in a P2P network, heterogenous and distributed MDCs are more appropriate. This is the problem which was addressed in [11]. The performance of wavelet-based MDC in peer to peer networks is investigated in [12].

While peer to peer networking has been explored by many researchers, the application of ant-based algorithms in P2P networks is rather a new concept. Although the ant metaphor has been successfully applied to routing of data packets both in wireless and fixed networks, little is yet known about its applications in peer to peer networks. Babaoglu et al. [14] has done a comprehensive research about the applicability of biological processes to distributed environments, including a discussion of ant-based methods in context.

SemAnt [16] suggests an ant-based query routing algorithm for P2P networks. The experimental approaches show that the algorithm produces robust results and converges fast. The author has extended the algorithm to include strategies for self-adaptation to volatile networks where nodes may leave or join at any time.

MUTE [18] is an ant-based peer to peer file sharing application. Its performance is investigated in [17]. It considers the case of network dynamism and its effects on query routing performance.

Anthill [19] is an open source framework for the design, implementation, and evaluation of ant algorithms in peer-to-peer networks.

TABLE I
PHEROMONE TABLE

Destination	Neighbor			
	a_1	a_2	...	a_m
d_1	p(1,1)	p(1,2)	...	p(1,m)
d_2	p(2,1)	p(2,2)	...	p(2,m)
$\vdots$	...	...	...	$\vdots$
d_n	p(n,1)	p(n,2)	...	p(n,m)

All of these efforts are dedicated to the design of query routing algorithms in peer to peer networks. To the best of our knowledge, this is the first work on ant-based rate allocation for media streaming in peer to peer networks.

III. AN INTRODUCTION TO ANT-BASED ROUTING ALGORITHMS

A. Foraging Behavior of Ants

Ants establish the shortest path between food and their nest in a fully-distributed and autonomous fashion. Ants first wander to search for food. When they find a food source, they return to the nest while leaving a trail of chemical substance called pheromone on their way back. The pheromone attracts other ants and guide them to the food. The probability that the ants coming later choose a path is proportional to the amount of pheromone on the path.

Although pheromone evaporates and decays, the following ants also leave additional pheromone and thus reinforce the path. If more ants travel over a path, they deposit more pheromone on it. Inversely, larger amounts of pheromone attract more ants. This behavior results in a positive feedback effect. Since the number of ants that complete their journey to the food source in a given time is larger on a shorter path than on a longer path, a shorter path can accumulate more pheromone and attract more ants.

At last, the shortest of alternative paths is selected. Most of ants take this shortest way to the nearest food source. Longer paths are also maintained because some ants are attracted to them by remnants of pheromone.

B. Ant-based Routing Algorithm

Based on these principles, an ant-based routing algorithm for communication networks was first suggested by Dorigio et. al. [20]. In the proposed algorithm, each node has a sort of routing table called pheromone table (See table I). Every node has an entry (row in pheromone table) for each possible destination in the network, and every table has an entry for each neighboring node. The (i,j) member of the pheromone table indicates the pheromone intensity between the current node and its jth neighbor for destination i. Pheromone table determines the routing behavior of nodes. When an ant arrives at a node, it chooses its next hop based on pheromone intensities of pheromone table in a probabilistic manner. This is in contrast to deterministic routing of other routing algorithms. Equation (1) [13] gives the probability with which an ant destined to node i, chooses node j as its next hop. Here, node

j is one of the neighbors of node k where the ant is currently visiting. This is called random proportional transition rule.

$$p_k(i,j) = \begin{cases} \frac{\phi_{ij}}{\sum_{l \in \Im} \phi_{il}}, & \text{if } i \in \Im; \\ 0, & \text{if } j \notin \Im; \end{cases} \quad (1)$$

In this equation, $\Im$ is the set of node k's neighbors that the ant has not visited yet and ϕ_{ij} is the (i,j) element of the pheromone table, indicating the pheromone intensity on link (k,j) for all ants destined to node i.

Each ant saves the path it travels so that when the destination is found, it could find its way back to the source. In its path backward to source, the ant updates pheromone tables of all nodes it has visited in the forward path according to the defined rules. Equation (2) indicates the updating rule.

$$\phi_{ij}(t+1) = (1-\alpha) * \phi_{ij}(t) + \Delta\phi \quad (2)$$

Here, $\Delta\phi$ is the amount of pheromone added to the link and α indicates pheromone evaporation. The amount of $\Delta\phi$ depends on network parameters such as delay and congestion. At the beginning of the algorithm, all links are initialized with a little amount of pheromone, $\phi_{ij}(0)$. Subsequently, the pheromone is updated according to equation (2) in each iteration.

C. Why Ant-Based Algorithms Are Appropriate for P2P Networks?

The simple ant colony optimization meta-heuristic shown in the previous section illustrates different reasons why this kind of algorithms could perform well in peer to peer networks.

Ant-based routing algorithms have several properties that make them an appropriate choice for peer to peer networks. More specifically, They are:

- Simple
- Purely distributed
- Multipath, which is an important and desirable property in unstructured peer to peer networks.
- Remarkably adaptable to network changes. When the shortest path is accidentally broken, a longer alternative path eventually becomes the new shortest path.

Of course ant-based algorithms have some disadvantages as well. For instance one of the biggest disadvantages of ant-based routing algorithms arises in highly dynamic networks where a large number of control packets is needed. In such cases, ant based algorithms need a large amount of overhead before they find the shortest path.

IV. ANT-BASED RATE ALLOCATION

In this section we provide a detailed explanation of our ant-based rate allocation algorithm for media streaming over peer to peer networks.

A. Problem Formulation

We model the peer to peer network as a connected graph G(V,E) where V is the set of nodes and E is the set of links. It is assumed that there are a number of peer nodes that can provide media contents. Those nodes that are willing to receive

a stream, send their requests to these nodes. Source nodes start media streaming based on user requests and network conditions.

In general, peers do not know the address of media providing nodes. As a result they search for these nodes across the peer to peer network. In this paper, we assume that a node has done the search and already knows the address of media providers (designing an ant based search algorithm is a subject of our future research). At this point the only problem is how to define rates according to which the destination node receives media streams from different media providers.

Assume that a node D has requested a media stream with rate $\Re$. It has found a group T of nodes that provide the media stream. Now the problem is how to allocate rate according to which each node sends the requested stream. In the simplest case, the set T contains only one node and this node has sufficient available bandwidth to support the required rate, $\Re$. In practice, such a node usually does not exist. Even if there is such a node, it soon becomes congested. Therefore, we usually have many nodes neither of which could support the required rate alone. It is the node D that specifies the rate of streaming from each server. We suggest an ant-inspired algorithm to do this rate allocation.

B. Ant-inspired Rate Allocation

We assign a pair (R, A) to each media provider where (R_i, A_i) represent the rate and availability of node i respectively. R_i is the rate that node i claims it can support. The availability of node i, represented by A_i, is a measure of time during which node i is on. The set T contains all the nodes that have the requested stream available.

In the first step, the node D initializes its pheromone table. Then according to its desirable rate, $\Re$, it generates a number of agents, which we call ants. Now we have a network in which the source has a direct link to each receiver. Here, by the *source* we mean the source of ants and in the same fashion, receivers are media providers who receive forward ants. The only thing to do is to assign pheromone to links. Having done that, ants follow the links with more pheromone in a probabilistic manner.

We add pheromone to links according to the rate and availability of the media provider at the end of the link. To be more precise, pheromone on link i is proportional to:

$$\Delta\phi = A_i * R_i \tag{3}$$

Assume that the node D has generated N forward ants. The number of ants is proportional to $\Re$. In other words, the number of forward ants that node D generates is proportional to the number of descriptions it requires. Since forward ants choose their links probabilistically, in the first step the same number of ants choose each link.

Each node receives a number of forward ants. The more ants a node receives, the higher rate it is asked to provide. In response to receiving forward ants, the node generates some backward ants and sends them back to D. The number of backward ants generated by node i is proportional to the

rate that the node could provide, i.e. R_i. It must be noted that backward ants do not carry traffic themselves but they represent the rate by which traffic is transmitted to D.

On receiving these backward ants, D updates its pheromone table according to equation (2), where $\Delta\phi$ is given by equation (3). It is clear that the pheromone values for those nodes that could provide higher rates during longer periods of time grow rapidly.

In the next steps, forward ants choose the nodes that have higher rates due to the pheromone update rule. In a few steps, forward ants find the best set of nodes that could provide the desired rate.

C. Algorithm Properties

The suggested algorithm has several desirable properties some of which are listed below:

- The algorithm is adaptable to congestion. To see this, assume that a node i has a high rate R_i and it is available any time so that A_i equals 1. Thus, the pheromone intensity for this node is so large that all ants choose this node with high probability. This will result in congestion on the path between D and i. Therefore, only a fraction of backward ants will be received by D and the pheromone increment on that link is less that what it could be if all backward ants were received by D. Due to pheromone evaporation, the excess pheromone on this link will be evaporated and the pheromone value for this node decreases eventually. Therefore, fewer ants will choose this node in next steps and less pheromone will be added to the link. Finally, the network reaches stability where there is no congestion on the links. This could be verified with simulation results of next section.

- Another property of the proposed algorithm is that it smoothly adapts itself with network changes. As an example, assume that node i fails. This means that it could not provide rate R_i temporarily. As a result, it will not send backward ants to D. Therefore, no pheromone will be added to its link and fewer forward ants will be sent to this server. Eventually, pheromone will evaporate and none of the forward ants would follow the link to this node. If in the middle of this process, node i is fixed, it will generate backward ants and its pheromone will rise again. Thus, forward ants will consider this node as a potential media provider again.

  According to equation (1), when the pheromone value for one neighbor is reduced, the probability that this node is chosen is reduced as well. Hence, the probability that other nodes are chosen increases. Therefore, whenever a node fails, other nodes compensate this failure in a smooth fashion. This proves the adaptability of our algorithm to network dynamism which is specially important in peer to peer networks.

  There is a tradeoff between the evaporation rate and the speed of adaptation to network changes. If the evaporation rate is high, the algorithm rapidly adapts itself to the changes in the network. But this changes may

be temporary. Thus, having a high evaporation rate is not desirable in all cases. The coefficient α shown in equation (2), which determines evaporation rate, should be adjusted carefully to cover all these scenarios. This adjustment could be done by the user according to network conditions. If network changes frequently and these changes are not temporary, it is better to have a high evaporation rate. However, if the rate of change is relatively low and most of changes are temporary, it is better to increase α so that the algorithm handle these changes smoothly without over reaction.

- Another advantage of our algorithm is its ability to handle inaccuracies in rate measurements. There could be cases where the actual rate of node i is not what it claims. This could happen either because node i's rate is not measured with high accuracy or there are bottlenecks in the physical path between node D and node i. In such cases, since the actual rate is less than what node i claims to be, the rate at which backward ants are received by D are less than what it should be. As a result, the amount of pheromone added to link (D, i) is less that its maximum. Thus, in next steps fewer forward ants are sent over this link and more forward ants are sent toward other nodes. This means that the rate at which D receives media stream from node i is less than its announced rate, either due to congestion in physical path or inaccuracies in rate measurements. It is obvious that to overcome congestion as mentioned above, we do not need any information about network topology. This is a great advantage in comparison to algorithms that need to know network topology to handle congestion, like PROMISE [5] or IPROMISE [7].

- In cases that one node fails and the rest of nodes could not provide additional rate to compensate the failure of other servers, it is needed to add other nodes to set T. These nodes are those which were found in the search phase. If there are no nodes to be added to the list, it is necessary to do a search phase again.

V. SIMULATION RESULTS

In this section, we investigate the performance of our algorithm. We examine different scenarios to show the advantages of our ant-based rate allocation compared to other rate allocation algorithms.

We have simulated our scenarios on the same topology that was used in [5] to show how PROMISE chooses the best set of nodes in a peer to peer network according to their rate and availability. While PROMISE needs to know the topology completely to choose the best set of nodes, our algorithm does not use any topology related information to do rate allocation. This topology is shown in figure 1. It is assumed that we have 6 media provider nodes. The numbers besides each node indicates its announced rate and its availability. The capacity of each bottleneck link is shown as well. In all parts, we assume that the node D has requested a media stream with the rate ($\Re$) of 1Mbps. Media stream is sent to node D in 8 descriptions. Ant-based rate allocation does not need to know any topology
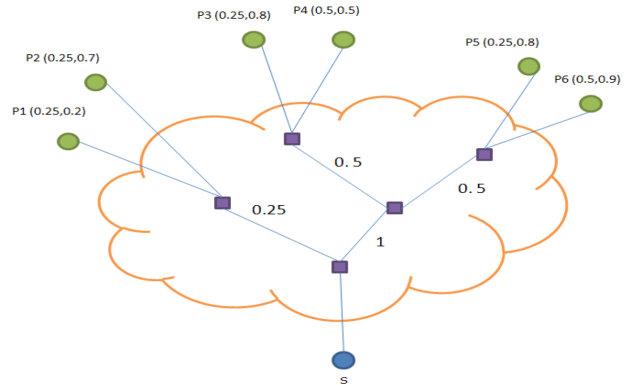

Fig. 1. Network topology

related information and exact available rate of each node to do rate allocation. As a result, the proposed algorithm is much less complex than other suggested algorithms. In addition it could easily handle the errors in claimed rates due to network congestion or lack of accurate measurements. Therefore, in the simulations we have assumed that our algorithm does not know the topology of the network. It only has access to the list of media providers with their claimed rate and availability, (R_i, A_i). We first simulate the process of acquiring the best set of nodes for rate allocation. Then, we show the adaptability of our algorithm to network changes which are quite regular in peer to peer networks.

A. Finding the Best Set of Servers

Topology-based rate allocation algorithms are able to estimate the maximum achievable rate of users in the network. To accomplish this task, they choose the best set of nodes that supports the required rate, $\Re$. However, due to uncertainties in the topology and rate of nodes, this choice may be inaccurate.

Our ant-based rate allocation does not need to know the exact rate or the availability of the node. We sort the nodes according to their availability and choose the first M nodes whose sum rate is greater than $\Re$. It must be noted that due to possible uncertainties in rate measurements, the actual sum rate may be less than $\Re$. But this is not a big issue since as we will see shortly, the algorithm overcomes this problem by gradually adding other nodes to the initial set until their sum rate is greater than $\Re$.

Based on this approach, nodes P_3, P_5 and P_6 are selected to provide the media stream (see figure 1). At this point, D starts to send forward ants and requests for media streams. The initial pheromone values on network links are shown in figure 2. In addition, figure 2 indicates the maximum achievable rate of media providers and the maximum achievable rate of ant-based rate allocation. From figure 1, it is obvious that the sum rate of media providers is only 0.75Mbps. This is because of the fact that nodes P_5 and P_6 share a 0.5Mbps link. This bottleneck prevents D from receiving its required rate of
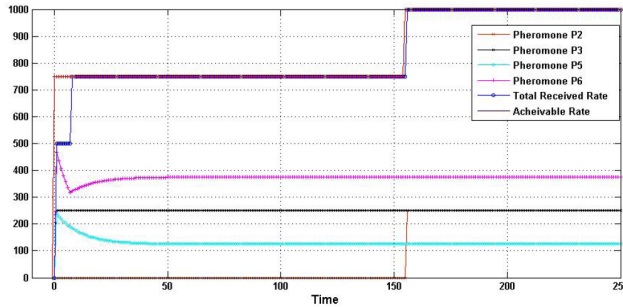
Fig. 2. Convergence of the algorithm to maximum achievable rate

1Mbps. It is shown in figure 2 that after a short period of time, the ant-based rate allocation reaches 0.75Mbps which is the maximum achievable rate in this case. It is clear from figure 2 that initially, D keeps sending requests based on 1Mbps. This means that D adjusted the number of forward ants based on 1Mbps. After a few steps, D adjusts the number of ants to represent 0.75Mbps. This is due to the fact that servers P_5 and P_6 generate backward ants according to their available rate, 0.5Mbps. But because of the bottleneck on the link with 0.5Mbps bandwidth, only a fraction of backward ants will be received by D. As a result, the pheromone added to links between D and nodes P_5 and P_6 is less than its possible maximum. At this point, node D decides that the chosen set of nodes do not have sufficient resources to support its required rate. Therefore, it selects another node from the sorted list of nodes and adds it to current set of media providers. In our case, node P_2 is added to the set of media providers. From figure 1 it is clear that the maximum achievable rate is now 1Mbps. Figure 2 indicates that our ant-based rate allocation algorithm will achieve this rate after addition of P_2. This clearly shows the adaptability of our algorithm without any knowledge about network topology. The rate of adaptation could be changed by adjusting the pheromone evaporation rate. We discuss this case shortly.

Now assume that because of congestion or lack of accurate measurements, we have 10% error in measured rates. If we use PROMISE [6] on the topology shown in figure 1, nodes P_2, P_3 and P_6 would be chosen as the set of media providers. But the actual rate of P_6 is 0.375Mbps rather than the measured value of 0.5Mbps. Thus, the maximum achievable rate would be 0.875Mbps rather than 1Mbps. As a result, the set of selected nodes could not support the required rate and all the efforts that PROMISE does to find the best set is futile. Therefore, it has to add another node, P_5. The final set, which is P_2, P_3, P_5 and P_6 is the same as the set that our ant-based algorithm finds, albeit with much less effort.

To summarize, if we have complete and accurate information about network topology and rate of nodes, algorithms like PROMISE [6] will find the best set of nodes for initiation of multimedia streaming. The negative point is that they are complex. But if there is uncertainty in measured values, and if there are no information about network topology, which is the case most of the times, our ant-based rate allocation method has a better performance.

B. Adaptability to Network Changes

One of the most important advantages of ant-based rate allocation is its smooth adaptation to network changes. Since most of P2P networks are unstructured, there are not many nodes with high rate and long availability like servers, i.e. with $A_i = 1$. Moreover, network topology is very dynamic and network changes quite often. Thus, any algorithm designed for peer to peer networks must cope with network dynamism. In this section, we consider scenarios in which changes occur in network and examine the response of our algorithm to these changes. We have run all simulations over the topology shown in 1.

In a P2P network, rate of nodes vary to a great extent. These variations could be temporary, long lasting or permanent. Any algorithm which is to be used in peer to peer networks should handle both of these changes. This is one of difficult challenges that algorithm designers have to deal with.

It is mentioned in [9] that a low pass filter should be applied to network changes so that temporary variations is separated from permanent ones. From one perspective, ant based algorithms implement a natural low pass filter. Since ants react to changes gradually, they completely discard a solution only if the duration of change is long enough. During this period, all of the pheromones for the invalid solution are completely evaporated. The rate of adaptation could be adjusted by carefully determining the evaporation rate of pheromone. This parameter could be chosen according to network conditions.

To clarify this, consider the case shown in figure 3. All of the conditions are the same as before. At t = 500 [1], P_2 loses 50Kbps of its rate. This means that the rate of this link should be decreased to 125Kbps because we have to send one description instead of two descriptions [2]. As a result, the amount of pheromone for this link will not increase as much as before. As we discussed in previous sections and from equation (1), the probability that this node is chosen by forward ants will be reduced and other nodes should compensate this loss of rate. However, since all nodes are sending media streams up to their rate limits, this loss could not be compensated. Therefore, the pheromone of other users will not increase (figure 3). At t = 600, network is recovered. If D had considered this variations as a permanent one, it would have added another node to the media providers set. It is clear that it does not add another node because this change is transient. When P_6 goes back to its normal conditions, gradually the pheromone of its link is increased and the rate of

---

[1]We have normalized time axis by the intervals of forward ants generation. Node D generates forward ants in regular intervals to maintain media streaming.

[2]Since we have 8 descriptions for a 1Mbps stream, each description has a rate of 125Kbps
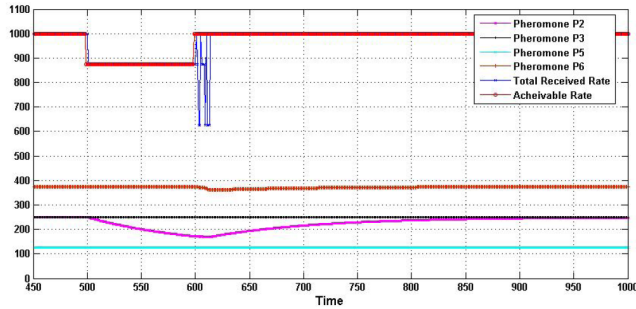
Fig. 3.   Adaptation of rate allocation algorithm to transient rate variations
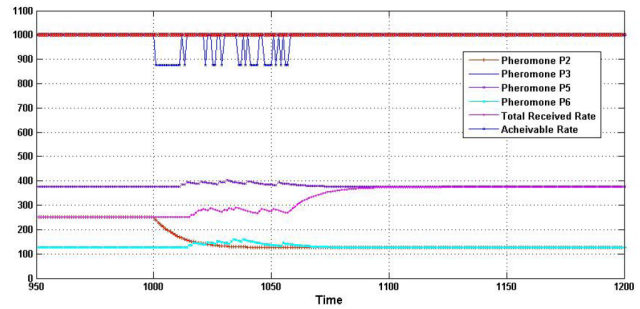

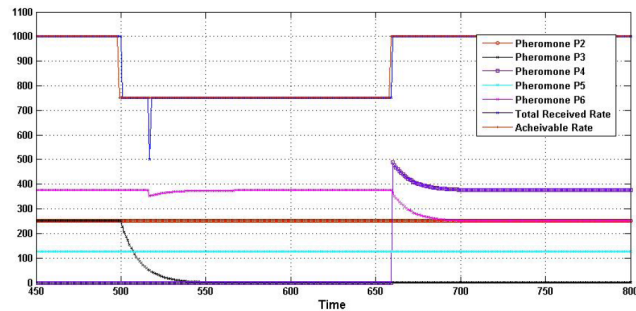Fig. 5.   Compensation of rate loss by other nodes

Fig. 4.   Adaptation of rate allocation algorithm to permanent node failures

media stream rises to its required rate. The rate of pheromone evaporation must be adjusted such that the pheromone is not evaporated in 100 time units, which is the duration of variations. In the simulated case, α is set to 0.01. In other cases, it is set to 0.1.

Now assume another case where P_3 fails permanently. This could occur if P_3 leaves the network for example. In this case, maximum achievable rate decreases to 0.75Mbps. Since the failure lasts for a long time, all of the pheromone on link to P_3 will finally evaporate and D will add another node, P_4, to the set of media providers. Since the initial amount of pheromone on the link toward P_4 is large, the rate of media stream from P_4 will be high. Therefore one of the descriptions of P_6 is assigned to P_4 and network converges to the stable solution. In figure 4 we have shown pheromone intensities for P_3, P_4 and, P_6.

In figure 4, the maximum achievable rate is 1.25Mbps and we use only 1Mbps of it. Now assume that P_2 loses 125Kbps of its rate permanently. Since the sum rate of nodes is still higher than 1Mbps, there is no need to add a new node and the rest of the nodes compensate the loss of rate. This is because of the fact that when P_2 reduces its rate, its pheromone will decrease gradually due to pheromone evaporation. According to equation (1), the loss of pheromone of one node will decrease the probability that this node is chosen by forward

ants and will increase the probability of choosing other nodes as the destination of ants. As a result, the rate of media streaming from these nodes will increase while the rate of P_2 will decrease. In the simulated case (shown in figure 5), node P_6 will increase its number of description from 2 (in previous cases) to 3. This will compensate the loss of rate and keep the media streaming rate at its desired value, 1Mbps.

Small variations of rate in figure 5 is because all the other nodes could compensate the loss of rate due to P_2 failure. But after a few steps, node P_6 is determined to do the compensation and other nodes continue to send their media stream as before.

Finally, consider the network condition shown in figure 3 that has reached stability. In this case, the maximum achievable rate is 1Mbps and there is no excess rate since the required rate by D is also 1Mbps. Now assume that P_6, whose *actual* (and not measured) rate is 450Kbps, for some reason reduces its rate to 250Kbps. Figure 6 illustrates the pheromone intensities for P_5 and P_6. It could be seen that while the total capacity of the network was 1Mbps, rate reduction of 200Kbps by P_6 does not change the rate of media stream received by D. P_5 compensate this loss shortly after the rate reduction, since it was not working at its limits due to the bottleneck on the links. When P_6 decreases its rate, P_5 increases its own rate. This scenario indicates that network changes in peer to peer networks do not deteriorate the performance of our algorithm in all cases. The suggested algorithm could easily adapts itself to such changes.

We note that the adaptation time of our algorithm can be varied by changing the parameter α and it can be determined by noticing the state of the nodes of P2P network. In addition, the minimum meaningful rate for our algorithm is the sufficient rate for transmission of one description of MDC and all rates below this value are equivalent to zero in our algorithm.

VI. Conclusions and Future Works

We have propose a new algorithm based on ant-colony optimization metaheuristic to do rate allocation in multimedia streaming P2P networks. The suggested algorithm does not
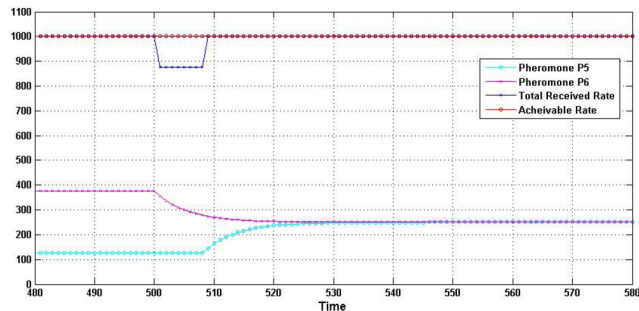
Fig. 6. Adaptation of rate allocation algorithm to permanent rate loss when there is a bottleneck in the network

need any topology related information to perform its tasks. Moreover, it could overcome inaccuracies and uncertainties in estimated network state information such as rate of nodes that provide media streams. We demonstrated that our algorithm will reach the maximum achievable rate of the network quite fast and with relatively little overhead.

Our algorithm is capable of handling network dynamism which is an inherent property of peer to peer networks. The suggested algorithm has an intrinsic low pass filter which separates transient network changes from permanent ones. If the changes in the network is transient, the algorithm compensates the temporary losses without much complexity. In cases where the network changes last longer, the algorithm overcomes losses by employing other nodes that have the media stream available. The rate of adaptation could be adjusted according to network conditions.

Our future works will include designing an ant-based search algorithm to find the nodes that provide media stream across the P2P network. Another important task would be modifying the proposed algorithm such that the pheromone evaporation rate adapts itself to different network conditions. This would be a great step forward since the resulted algorithm will adapts itself to almost any network condition.

Extension of our algorithm to P2P networks with multiple streaming sessions is another task which is to be fulfilled. Finally, implementing the ant-based rate allocation algorithm in a real world test-bed will be another important aspect that can be pursued in future.

VII. ACKNOWLEDGMENT

REFERENCES

[1] Y. Cui, B. Li, and K. Nahrstedt, oStream: asynchronous streaming multicast, IEEE J. Select. Areas in Comm., vol. 22, Jan. 2004.

[2] D. A. Tran, K. A. Hua, and T. T. Do, A peer-to-peer architecture for media streaming, in IEEE J. Select. Areas in Comm., vol. 22, Jan. 2004.

[3] S. Itaya, T. Enokido and M. Takizawa, A High Performance Multimedia Streaming Model on Multi-Source Streaming Approach in Peer-to-Peer Networks, Proceeding of 19th International Conference on Advanced Information Networking and Application (AINA'05), pp. 27-32, Mar. 2005.

[4] S. Itaya, T. Enokido and M. Takizawa, A Scalable Multimedia Streaming Based On Multi-Source Streming Concept, Proceeding of 11th Conference on Parallel and Distributed Systems, pp. 15-21, Jul. 2005.

[5] M. Hefeeda, A. Habib, D. Xu, B. Bhargava, and B. Botev, "CollectCast: A peer-to-peer service for media streaming," in ACM Multimedia'03, 2003.

[6] M. Heffeeda, A. Habib, B. Botev, D. Xu, and B. Bhargava, PROMISE: peer-to-peer media streaming using CollectCast, in Proc. ACM Multimedia (MM03), Berkeley, CA, Nov., 2003.

[7] M. H. Firooz, K. Ronasi, M. R. Pakravan, A. R. Nasiri-Avanaki, "IPROMISE: REliable Multi-sender Algorithm for Peer to Peer networks", in Proc. IEEE 2'nd international conference on communication system software and middleware (COMSWARE), Nabglore, India, Jan. 7-12, 2007.

[8] M. H. Firooz, K. Ronasi, M. R. Pakravan, A. R. Nasiri-Avanaki, "A Multi-Sender Multicast Algorithm for Media Streaming on Peer-to-Peer Networks", Computer Communications 30 (2007) 21912200

[9] M. Mubashar, A. Toufik, M. Djamal-Eddine and Meddour, D. "Adaptive packet video streaming over P2P networks. In Proceedings of the 1st international Conference on Scalable information Systems", Hong Kong, May 30 - June 01, 2006.

[10] R. Rejaie and A. Ortega, PALS: peer to peer adaptive layered streaming, in Proc. NOSSDAV03, Monterey, CA, USA, Jun. 2003.

[11] X. Xu, Y. Wang, S. S. Panwr, K. W. Ross, "A Peer-to-Peer Video on Demand System Using Multiple Description Coding and Server Diversity", Proceeding of ICME05, pp:1-4, Jul. 2005.

[12] J. Kim, R. M. Mersereau, Y. Altunbasak, "Distributed Video Streaming Using Multiple Description Coding and unequal Error Protection", IEEE Tran. On Image Processing, Vol. 14, No. 7, pp. 849-861, Jul. 2005.

[13] M. Dorigo and G. D. Caro. New Ideas in Optimization, chapter The Ant Colony Optimization Meta-Heuristic, pages 1132. McGraw-Hill, 1999.

[14] O. Babaoglu, G. Canright, A. Deutsch, G. Di Caro, F. Ducatelle, L. M. Gambardella, N. Ganguly, M. Jelasity, R. Montemanni, and A. Montresor. Design Patterns from Biology for Distributed Computing. In Proceedings of the European Conference on Complex Systems, 2005.

[15] Myrmedrome: A Real Ant Colony Simulator, $http://www.not-equal.eu/myrmedrome/main_en.html$

[16] Elke Michlmayr, Arno Pany, Sabine Graf: "Applying Ant-based Multi-Agent Systems to Query Routing in Distributed Environments," Proceedings of the 3rd IEEE Conference On Intelligent Systems (IEEE IS06), London, UK, September 2006.

[17] M. Ciglaric, T. Vidmar, "Ant-inspired query routing performance in dynamic peer-to-peer networks," ipdps, p. 287, Proceedings 20th IEEE International Parallel and amp; Distributed Processing Symposium, 2006

[18] MUTE: Simple, Anonymous File Sharing, http://mute-net.sourceforge.net/

[19] O. Babaoglu, H. Meling, and A. Montresor. Anthill: A Framework for the Development of Agent-Based Peer-to- Peer Systems. In Proceedings of the 22nd International Conference on Distributed Computing Systems, 2002.

[20] G. D. Caro and M. Dorigo. AntNet: Distributed Stigmergy Control for Communications Networks. Journal of Artificial Intelligence Research, 9:317365, 1998.