

Maximizing Open Capacity in Mobile Optical Backbone Networks Using Controllable Mobile Agents

Fangting Sun, Abhishek Kashyap, Mark Shayman
Department of Electrical and Computer Engineering
University of Maryland, College Park, MD 20742
Email: {ftsun, kashyap, shayman}@eng.umd.edu

Abstract—We consider a mobile backbone network with free space optical point-to-point links. Requests for aggregate bandwidth between pairs of backbone nodes arrive one-by-one, and a bandwidth guaranteed connection is established if there are sufficient network resources; otherwise, the request is rejected. In addition to the ordinary backbone nodes, there are a limited number of controllable mobile agents; these are nodes that may be located as desired to optimize the performance of the network. The specific problem we consider is that of determining the positions for the mobile agents to maximize the minimum maxflow over all source-destination pairs. We show that this problem is NP-hard. We then develop a heuristic algorithm to locate the agents based on the current network state. It is shown that by using the algorithm to strategically place a limited number of agents, a significant improvement in network performance can be achieved.

I. INTRODUCTION

A mobile ad hoc network (MANET) is a group of mobile nodes without requiring centralized administration or fixed network infrastructure, in which the mobile nodes can communicate with other nodes directly or through cooperatively forwarding packets for each other. Recently a wide range of military and commercial applications have shown interest in MANETs since they can be easily deployed and reconfigured. However, it has been shown that a flat ad hoc network has poor scalability [1]. To build a large-scale MANET, a promising solution is to organize nodes in a hierarchical way [2], [3]. That is, some nodes are selected to form a higher level network called the backbone network, in which the backbone nodes can establish links among themselves.

Free-space optics (FSO) technology is expected to deliver unprecedented bandwidth, massive carrier reuse, ultra-low inter-channel interference, low power consumption, and cost savings where electrical wires and optical fibers are too expensive to deploy and maintain [4]. These characteristics make them more suitable for wireless backbone networks than RF and wireline optical links, since RF links do not offer the high bandwidth required for backbone networks, while wireline optical links do not support mobility.

In this paper, we focus on the optical backbone network in a MANET where each backbone node can build point-to-point free space optical (FSO) links with other backbone nodes

within its transmission range. However, similar results would be expected for highly directional radio frequency (RF) links. In a MANET, nodes with low mobility are more eligible to act as backbone nodes, so we assume that the backbone nodes have limited mobility. Consequently, it is reasonable to set up bandwidth guaranteed connections (e.g., MPLS label switched paths) to carry aggregate traffic, which has the advantage that QoS provisioning can be supported on a scalable basis.

Upon a bandwidth guaranteed connection setup request arrival, the connection will be set up if there are enough resources (link bandwidth and node transceivers), else it will be rejected. To reduce the ratio of rejected requests due to insufficient network resources, researchers suggest some algorithms to load-balance traffic in the networks. The widest-shortest path algorithm (WSP) [5] finds a feasible min-hop path between source and destination such that the chosen path has the maximum bottleneck link residual capacity among all shortest paths. If the source-destination pairs are known, minimum interference routing algorithms [6], [7], [8] will find a route that does not “interfere too much” with a route that may be critical to satisfy a future demand. [9] extends the minimum interference concept to topology control and routing in free space optical networks by considering both critical transceivers and critical links instead of only critical links.

Another way to reduce the ratio of blocked requests is adding more resources to the network. In this paper, we propose addition of controllable mobile agents to congested areas of the network to alleviate the resource shortage. Controllable mobile agent is first suggested in [10] to help restore the broken bandwidth guaranteed connections in backbone networks. In this paper, the mobile agents will help carry all traffic instead of just broken traffic. The agents have the same transmission capability as the general backbone nodes. The difference is that the movement of the agents is controllable, and they can be adaptively relocated based on the network status; the movement of general backbone nodes is random and not predictable.

The following scenario is considered in this paper. There is a set of mobile backbone nodes and controllable mobile agents, which are geographically distributed in an area. The backbone nodes may be clusterheads, and they will move when their clusters relocate. We assume the backbone nodes move randomly inside the network area with low mobility. A

subset of backbone nodes constitutes sources and destinations. Requests for aggregate bandwidth between pairs of backbone nodes arrive one-by-one and future demands are unknown. For each request, if sufficient resources are available, a bandwidth guaranteed connection is established; otherwise, the request is rejected. If an established connection fails, the source tries to find a new route, if such new route is not available, then the request is rejected. In order to reduce the ratio of rejected requests, the mobile agents should be placed in an efficient way, and be periodically relocated based on the dynamically changing network status. We try to place the agents in the network to maximize the minimum maxflow among all source-destination pairs. However, it is shown that this problem is NP-hard, thus we propose a heuristic algorithm. It is shown that by using the algorithm to strategically place a limited number of agents, a significant improvement in network performance can be achieved.

The rest of paper is organized as follows: Section II describes the system model and basic assumptions. The agent-location problem is formulated in III and efficient schemes are described in section IV. Section V presents the simulation methodology and simulation results. Finally, section VI concludes this paper.

II. SYSTEM MODEL

In this section, we specify our assumptions about optical backbone network, node movement, bandwidth guaranteed connection request arrival process and the procedure for their creation and deletion.

A. Optical backbone network

There are two types of nodes in a wireless optical backbone network: mobile backbone node and controllable mobile agent. Each node is equipped with a limited number of FSO transmitters and receivers, and an RF transceiver. In the backbone network, high bandwidth data traffic is transmitted using the FSO transceivers, and the lower bandwidth control messages are transmitted by RF transceivers. Also the RF transceiver is used to collect the traffic within the cluster. A unidirectional FSO link can be set up between a pair of FSO transmitter and FSO receiver within each other's transmission range. We say there exists a *potential link* from node A to node B if they are in each other's transmission range, and we say there exists an *actual link* from node A to node B if they are in each other's transmission range and the FSO link has been set up from A to B. Furthermore, only one actual link is allowed between two nodes. Physically, to set up an actual FSO link between two nodes, the following procedures are needed: pointing, acquisition and tracking, which can be performed quickly [4].

B. Node movement

There are N backbone nodes distributed in a 2-dimensional plane denoted by A . Each backbone node moves randomly according to the *random waypoint model* [11]: a node starts at a random position in the network, waits for a duration called the *pause time*, which is modelled as an exponentially distributed random variable with mean μ , then randomly

chooses a destination location and moves towards it with a velocity uniformly chosen between v_{min} and v_{max} . When it arrives at that location, it waits for another random pause time and repeats the process.

There are M controllable mobile agents in the network and we have full control of their mobility. We can think of the mobile agents as Unmanned Aerial Vehicles (UAVs). We assume agents can be relocated instantly.

C. Request arrival and departure process

In the network, a backbone node will collect the traffic demands within its cluster and issue requests for aggregate traffic to other backbone nodes. A subset of backbone nodes constitutes sources and destinations. We assume the traffic arrival process at each source node is a Poisson process with rate λ . The agents do not generate or terminate any traffic. Each request has a random duration which is modelled as an exponential random variable with mean τ . After the holding time, the request leaves the network.

D. Creation and deletion of bandwidth guaranteed connection

Creation of a bandwidth guaranteed connection for a request is performed in two steps. In the first step, the source of a request performs route discovery to find a valid route with enough resources to the destination of the request. This means that (1) for each actual link in the route, the available link bandwidth is at least as great as the bandwidth requested; (2) for each potential link in the route, there is an available transmitter at the head node of the link and an available receiver at the tail node of the link. The route discovery can be achieved by using the existing mobile ad hoc network routing protocols, such as DSR [11] with some modification, and is usually executed through RF links. During this step, the position of each node in the route is also returned to the source. After the route has been successfully discovered, the second step is to set up the FSO links on the route and reserve necessary bandwidth. If a link in the route is a potential link, then the actual link should be set up through necessary pointing, acquisition, and tracking.

A reserved path will be released when the request expires. The bandwidth reservation along the route is released. Any link for which there is no remaining bandwidth reservation is torn down and the corresponding transmitter and receiver are freed up. Thus, the actual link becomes a potential link. Furthermore, upon a bandwidth guaranteed connection breakage, the source will release the corresponding resources along the old route and try to find a new route. In this paper, we only consider the connection breakage due to the node mobility.

III. PROBLEM FORMULATION

In this section we first give an informal description of the problem, and then a more formal mathematical description and also a proof of NP-hardness of this formulation. The NP-hardness justifies use of a heuristic algorithm in the absence of a known approximation algorithm.

To reduce the ratio of rejected requests, there are many approaches as stated in section I. In this paper, we propose

to put some agents into the network to improve the performance. Consider the maximum flow (max-flow) [12] value θ_{sd} between a given source-destination pair (s, d) . θ_{sd} is an upper bound on the total amount of bandwidth that can be routed between (s, d) . If θ_{sd} is large enough, requests between source-destination pair (s, d) will not be rejected in the near future. Since the future requests are not predictable, to avoid rejecting requests, the best solution is to keep maxflow values of all source-destination pairs large. Thus, by adding some agents into the network, we want to achieve the following objective

$$\text{maximize } \min_{(s,d) \in \mathcal{P}} \theta_{sd}$$

where \mathcal{P} is the list of source-destination pairs, θ_{sd} is the maxflow between (s,d) , i.e., the objective is to maximize the minimum maxflow among all source-destination pairs. We denote this problem as MAX-MIN-MAX.

In free space optical backbone networks, in addition to the bandwidth constraint and transmission range constraint, the limited number of the transmitters (receivers) is a constraint too. Furthermore, we regard the network as a static network when solving the problem.

Let $G(V, E)$ be the given network in 2D plane, where V is the set of the backbone nodes with $n = |V|$ and E is the set of links which connect the backbone nodes with $m = |E|$. Let \mathcal{A} be the set of the mobile agents with $a = |\mathcal{A}|$. Let (u^x, u^y) denotes the coordinates of node $u \in V \cup \mathcal{A}$. Notice, the coordinates of the backbone nodes are given, while we need to compute the coordinates for the agents. We assume the backbone nodes and the agents are numbered sequentially. Let B be the maximum bandwidth of the links, D be the maximum transmission range of nodes, and T be the maximum number of transmitter (receiver) interfaces one node can have. Then the given network can be fully described by $(G, \mathcal{A}, B, D, T, \mathcal{P})$. Let $M_{n \times m}$ represent the node-edge incidence matrix for graph G . Each row in this matrix corresponds to a backbone node in the graph and each column of the matrix corresponds to an edge. Each column has exactly two non-zero entries. The column corresponding to edge (u, v) has $a+1$ in the row u and $a-1$ in the row v and a zero in all other rows. We use another matrix X to represent the additional edges due to the addition of agents into graph G . X is an $(n+a) \times (2an+a(a-1))$ matrix, where the first n rows represent the backbone nodes and last a rows represent the agents, the $2an+a(a-1)$ columns represent all possible edges that have at least one adjacent node as agent, and are ordered in lexicographically. Let \bar{y}^{sd} be an $m+2an+a(a-1)$ vector corresponding to each pair $(s, d) \in \mathcal{P}$, where each entry corresponds to one edge and the value of each entry is the amount of traffic with source s and destination d that traverses that edge. Let θ_{sd} be a scalar that is the possible flow between s and d in the network. Let \bar{R} be an $m+2an+a(a-1)$ vector of residual capacities, where R_i is the residual capacity of edge i if $i \leq m$ and $R_i = B$ for all other entries. Let \bar{e}^{sd} be an $n+a$ vector with a +1 in position s and a -1 in position d . Let $dis(u, v)$ represent the

distance between nodes u and v . Define \bar{M} as follows:

$$\bar{M} = \begin{bmatrix} M & X_n \\ 0 & X_a \end{bmatrix}$$

where X_n consists of the first n rows of matrix X and X_a consists of the remaining rows of X . Let $\bar{\mathbf{1}}$ denote the all-one column vector with the dimension being $m+2an+a(a-1)$. Then this MAX-MIN-MAX problem can be stated as follows:

$$\begin{aligned} & \max z & s.t. \\ & \bar{M}\bar{y}^{sd} = \theta_{sd}\bar{e}^{sd} & \forall (s, d) \in \mathcal{P} \end{aligned} \quad (1)$$

$$z \leq \theta_{sd} \quad \forall (s, d) \in \mathcal{P} \quad (2)$$

$$0 \leq \bar{y}^{sd} \leq \bar{R} \quad \forall (s, d) \in \mathcal{P} \quad (3)$$

$$-\infty \leq u^x, u^y \leq +\infty \quad \forall u \in \mathcal{A} \quad (4)$$

$$dis(u, v) = \sqrt{(u^x - v^x)^2 + (u^y - v^y)^2} \quad \forall u, v \in V \cup \mathcal{A} \quad (5)$$

$$-dis(u, v) \cdot X_{u,uv} \cdot X_{v,uv} \leq D \quad u \in \mathcal{A} \text{ or } v \in \mathcal{A} \quad (6)$$

$$(\bar{M} \times \bar{M}^T)_{u,u} + (\bar{M} \times \bar{\mathbf{1}})_u \leq 2T \quad \forall u \in V \cup \mathcal{A} \quad (7)$$

$$(\bar{M} \times \bar{M}^T)_{u,u} - (\bar{M} \times \bar{\mathbf{1}})_u \leq 2T \quad \forall u \in V \cup \mathcal{A} \quad (8)$$

$$x_{ij} = 0, \pm 1 \quad \forall x_{ij} \in X \quad (9)$$

$$X^T X \leq 2 \quad (10)$$

$$X^T \bar{\mathbf{1}} = \bar{\mathbf{0}} \quad (11)$$

Equation 1 is the flow constraint for each source-destination pair in \mathcal{P} : the input traffic should equal the output traffic at every node except the source and destination. Equation 2 along with the objective maximizes the minimum maxflow. Equation 3 gives the constraint on bandwidth, Equations 4-6 give the constraints on transmission range, and Equations 7, 8 give the constraints on the number of transceivers. Equations 9-11 require that each column of matrix X is either all 0 or contains one +1, one -1 and the rest 0.

Now we prove that MAX-MIN-MAX is NP-Hard.

Theorem 1: The problem MAX-MIN-MAX is NP-Hard.

Proof:

Consider the problem's corresponding decision version MAX-MIN-MAX($G, \mathcal{A}, B, D, T, \mathcal{P}, t$): $G, \mathcal{A}, B, D, T, \mathcal{P}$ are the same as defined previously, and t is the amount of increase we want in the minimum maxflow using $|\mathcal{A}|$ agents. The problem is to check if the increase t is feasible by computing the locations of $|\mathcal{A}|$ agents. Given a set of positions to locate the agents, it is easy to check if the increase t is feasible in polynomial time, so MAX-MIN-MAX $\in NP$. Consider another problem STP-MSPBEL(G, D, k): $G(V, E)$ is a geometric graph, D is the bound on the edge length and k is the number of Steiner nodes. The problem is to check if the input graph can be connected using k Steiner nodes with the bounded edge length constraint. STP-MSPBEL(G, k) is NP-Complete [13]. To show MAX-MIN-MAX($G, \mathcal{A}, B, D, T, \mathcal{P}, t$) is NP-Complete, we need a polynomial time reduction from STP-MSPBEL to MAX-MIN-MAX.

Let (G, D, k) be the input for STP-MSPBEL, and suppose G has C connected components G_1, G_2, \dots, G_C . Let $G' = G$, select one node $s \in V(G')$ as source, and from each connected

component G'_i select one destination $d_i \in V(G'_i)$. So the source destination pair list $\mathcal{P}' = \{(s, d_i), i = 1, \dots, C\}$. Further, we set $D' = D, \mathcal{A}' = \{a_i, i = 1, \dots, k\}, B' = C, T' = |V(G')|, t' = C - 1$. Then $(G', \mathcal{A}', B', D', T', \mathcal{P}', t')$ is the input for MAX-MIN-MAX. It is easy to show that this reduction can be finished in polynomial time and STP-MSPBEL(G, D, k) is feasible if and only if MAX-MIN-MAX($G', \mathcal{A}', B', D', T', \mathcal{P}', t'$) is feasible. So MAX-MIN-MAX($G, \mathcal{A}, B, D, T, \mathcal{P}, t$) is an NP-Complete problem too.

Thus the problem MAX-MIN-MAX is NP-Hard. \square

IV. SOLUTION APPROACH

In this section we propose a heuristic solution to MAX-MIN-MAX problem in FSO networks.

At a high level, the basic idea of the agent-location scheme in mobile FSO networks can be described as follows. An agent collects the necessary information from all other nodes and then solves MAX-MIN-MAX heuristically to find the good position to locate itself at. All agents will find their positions independently. Since the network topology changes over time, the agents will relocate themselves periodically based on current network situation.

A. Locating the Agents

The objective of the algorithm is to find the positions for the agents to maximize the minimum maxflow for all source-destination pairs. According to the Max-Flow Min-Cut Theorem [12], for any source-destination pair, the max-flow value is equal to the min-cut value. If the value of the min-cut is increased, the value of the max-flow increases too. The min-cut of a pair (s, d) divides the network into two subsets S and $\bar{S} = V - S$ with $s \in S, d \in \bar{S}$. The min-cut is denoted by $[S, \bar{S}]$. Then the value of the min-cut for (s, d) is

$$MinCut(s, d) = \sum_{(i, j) \in (S, \bar{S})} u_{ij}$$

where u_{ij} is the residual bandwidth for edge (i, j) . So the value for $MinCut(s, d)$ can be increased by adding edges between S and \bar{S} . The key idea of our algorithm is to put the agents into the network in such a way as to provide additional edges for the min-cut of the source destination pair.

First, compute the min-cut for all source-destination pairs. Due to the hardness of the problem of finding the min-cut for a pair of nodes with transmission interface constraints, the min-cut is computed based on the actual network, where all potential links are neglected. Furthermore, during the computation, the computing agent will neglect the effect of itself, i.e., delete itself from the network graph. The min-cut for a pair of nodes can be calculated by the labeling algorithm in $O(mnU)$ time [12] where m is the number of the actual links, n is the number of nodes, and U is link capacity.

Second, order the min-cuts by increasing value as $\mathcal{L} = \{[S_1, \bar{S}_1], [S_2, \bar{S}_2], \dots, [S_{|\mathcal{P}|}, \bar{S}_{|\mathcal{P}|}]\}$. In this step, we want to find the minimum min-cut $[S_{min}, \bar{S}_{min}]$. Since the min-cut is computed based on the actual network, there might exist nodes $u_i \in S_i, v_i \in \bar{S}_i$ such that (u_i, v_i) is a potential edge

and u_i has a free transmitter interface and v_i has a free receiver interface. That means (u_i, v_i) can be turned into an actual link and then make an additional connection between min-cut $[S_i, \bar{S}_i]$. So $[S_1, \bar{S}_1]$ may not be the minimum min-cut due to the existence of potential links. We check $[S_1, \bar{S}_1]$ to see if there exist some potential links that can be transformed into actual links; if such potential links exist, change the potential links, which lie between S_1 and \bar{S}_1 , into actual links randomly until no such links exist. (Those links will be marked as actual links for computation purpose) Then recompute the min-cut for this source-destination pair and insert it back into \mathcal{L} , and then check the new $[S_1, \bar{S}_1]$ and so on; if no such potential links exist, $[S_1, \bar{S}_1]$ is the minimum min-cut we want. In worst case, this step may compute the min-cut $O(kn^2)$ times, where k is the number of available interfaces and n is the number of nodes. However, in most cases the minimum min-cut can be found quickly.

Finally, we try to locate the agent around the minimum min-cut $[S_{min}, \bar{S}_{min}]$. Enumerate the node pairs $\{(u_i, v_i), u_i \in S_{min}, v_i \in \bar{S}_{min}\}$ such that u_i has a free transmission interface, v_i has a free receive interface and the distance between u_i and v_i is smaller than $2D$ (D is the transmission range). For each pair (u_i, v_i) , compute the corresponding min-cut $MinCut_i$ based on $G \cup \{(u_i, v_i)\}$, and then select the maximum $MinCut_m$ among them and place the agent to connect (u_m, v_m) . If the node pair list is empty, delete $[S_{min}, \bar{S}_{min}]$ from \mathcal{L} and then repeat the above procedure.

The whole algorithm is described in Algorithm 1.

Algorithm 1 Heuristic algorithm for MAX-MIN-MAX

Input: A network graph $G(N, E)$. N is the nodes set including all nodes except the agent under consideration, E is the actual links set; a set \mathcal{P} of source-destination pairs.

Output: A position to locate the agent.

- 1: Compute the min-cut $[S_{sd}, \bar{S}_{sd}]$ for each pair $(s, d) \in \mathcal{P}$
 - 2: Order the min-cuts in increasing order into list $\mathcal{L} = \{[S_1, \bar{S}_1], [S_2, \bar{S}_2], \dots, [S_{|\mathcal{P}|}, \bar{S}_{|\mathcal{P}|}]\}$
 - 3: **if** (potential links can be changed to actual link between $[S_1, \bar{S}_1]$) **then**
 - 4: virtually transform such potential links into actual links randomly until no such link exists
 - 5: Recompute the min-cut $[S_1, \bar{S}_1]$ for pair (s_1, d_1)
 - 6: insert $[S_1, \bar{S}_1]$ back into \mathcal{L} and reorder \mathcal{L} , go to step 3
 - 7: **else**
 - 8: Enumerate set $\mathcal{C} = \{(u_i, v_i), u_i \in S_1, v_i \in \bar{S}_1, dis(u_i, v_i) \leq 2D\}$, s.t. each u_i has free transmission interface and each v_i has free receive interface
 - 9: **if** ($\mathcal{C} = \emptyset$) **then**
 - 10: $\mathcal{L} = \mathcal{L} \setminus \{[S_1, \bar{S}_1]\}$ go to step 3
 - 11: **end if**
 - 12: compute min-cut $MinCut_i$ based on $G \cup \{(u_i, v_i)\}$, $\forall (u_i, v_i) \in \mathcal{C}$
 - 13: find $MinCut_m = \max\{MinCut_i\}$, return the middle point of (u_m, v_m)
 - 14: **end if**
-

B. Dynamic Relocation of Agents

In the proposed scheme, the mobile agents are randomly allocated inside the area during the initialization procedure. Since the network topology and connection patterns change randomly with time, to maximize the minimum maxflow, the mobile agents should also be dynamically relocated according to the network status. After being allocated to the network, each mobile agent periodically collects the necessary information, including the node status and position and link status, from all other nodes through RF links and solves MAX-MIN-MAX.

V. PERFORMANCE STUDIES

A. Simulation Methodology

The simulation parameters are listed in Table I. We use a rectangular space of size 10000m \times 10000m. The total number of backbone nodes is 100, and the maximum FSO transmission range is 2000m. Each node has 4 FSO transmitters and 4 FSO receivers, and the FSO link bandwidth is 20 units per sec. For each traffic pair, the bandwidth guaranteed connection request arrival process is a Poisson process, and the average request interarrival time is chosen uniformly at random between 500 to 1000 sec. For each request, the duration is an exponentially distributed random variable with mean 2500 sec, and the bandwidth demand is uniformly distributed between 2 to 6 units per sec. When performing route discovery for a request, dynamic source routing (DSR) [11] is used, and the minimum-hop path satisfying the bandwidth and interface constraints is returned.

TABLE I
SIMULATION PARAMETERS

Dimensions of Space	10km \times 10km
Number of backbone nodes	100
Mobility model	Random waypoint
Maximum Velocity (v_{max})	20 m/s
Minimum Velocity (v_{min})	10 m/s
Number of FSO transmitters(receivers)	4
Maximum Transmission Range	2000 m
Link Bandwidth	20 units /s
Average request Inter-Arrival Time	500-1000 seconds
Average request duration	2500 seconds
Bandwidth demand	2-6 units /s

In the simulations, each backbone node moves randomly according to the random waypoint model and we set v_{min} to be 10m/s, v_{max} to be 20m/s, and different average pause time is used in different experiments.

In the simulations, the number of agents varies in different experiments, and are put into the network in a controllable way. For each agent, after staying at the same position for 5000 seconds, it will collect necessary information from other nodes and compute the next position it will move to or decide to stay at the same position. In the simulation, the RF transmission range is also 2000m. All experiments are run for 200,000 seconds. For each simulation setup, the results are averaged over 10 rounds, and at each round the random seed is changed to generate different initial topology, mobility pattern and

traffic pattern. Since random waypoint model is not stable at the beginning [14], we add 5000 seconds extra warmup time to each experiment.

B. Performance Evaluation

We first examine rejected request ratio and total network throughput under different situations. Fig. 1 shows the simulation results for the rejected request ratio and total network throughput under various number of agents and different average pause time when there are 40 source-destination pairs. From Fig. 1 we can see that the rejected request ratio decreases quickly and the total throughput increases quickly with the increase of agent number, especially when the number of agents is small. For example, when using 5 agents, the rejected request ratio is decreased by 29.7%, 30.5% and 30.2% when average pause time is 5000, 10000 and 15000 seconds respectively compared with the situation of no agent. And the total network throughput is increased by 13%, 10% and 8% when the average pause time is 5000, 10000 and 15000 seconds.

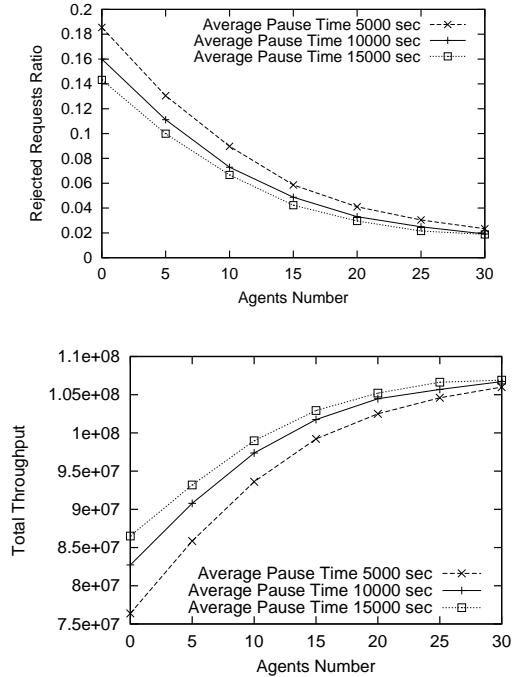


Fig. 1. The rejected request ratio and total throughput for different number of agent when there are 40 traffic pairs

From Fig. 1 it is also easy to see that there is little benefit from additional agents once the number of agents reaches 15. This is because when 15 agents are put into the network, most requests are already fulfilled, so there is little opportunity to further increase the performance by adding more agents.

We have also explored the performance when there are 60 source-destination pairs, which means more traffic in the network. The results for 60 pairs are shown in Fig. 2. According to this figure, similar to the case with 40 s-d pairs, the performance increases quickly when a few agents are added into the network and the increase slows down after more agents

are added. However, in this case, significant benefits accrue from additional agents until the number of agents reaches approximately 30. Thus, when the traffic load is higher, it is beneficial to use a larger number of agents.

Figures 1 and 2 show that when a limited number of agents is strategically placed according to the heuristic algorithm, a significant improvement in network performance can be achieved. However the performance benefit for each additional agent decreases as the total number of agents increases.

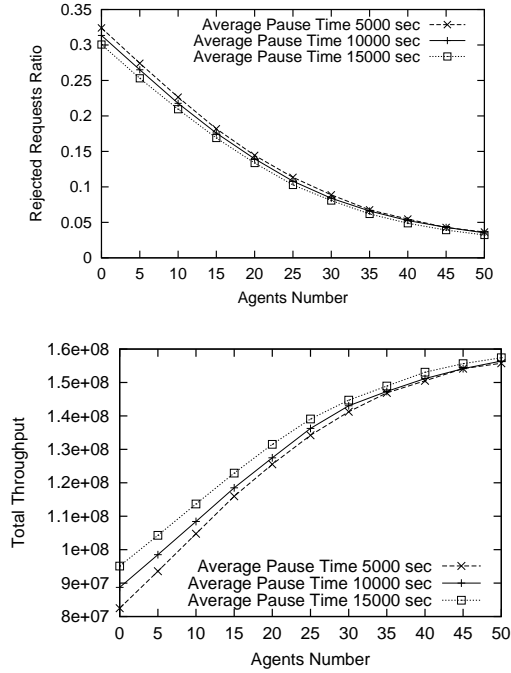


Fig. 2. The rejected request ratio and total throughput for different number of agent when there are 60 traffic pairs

Fig. 3 shows the comparison results under different agent setup. In Fig. 3, “mobile agents” denotes the setup where agents can be dynamically relocated, while “static agents” denotes the setup where agents are randomly distributed inside the area according to a uniform distribution and are not relocated during the whole simulation time. From Fig. 3 we can see that the effect of 10 static agents is worse than 10 mobile agents and similar to 5 mobile agents. Thus, network status-dependent relocation of agents performs much better than random location of agents.

VI. CONCLUSION

In this paper, we consider a problem of adding mobile agents to the network to maximize the minimum maxflow over all source-destination pairs. We give mathematical formulation of this problem and prove that this problem is NP-Hard. We then develop a heuristic algorithm to position the agents based on the current network state. Through extensive simulation experiments, we show that the proposed scheme works well.

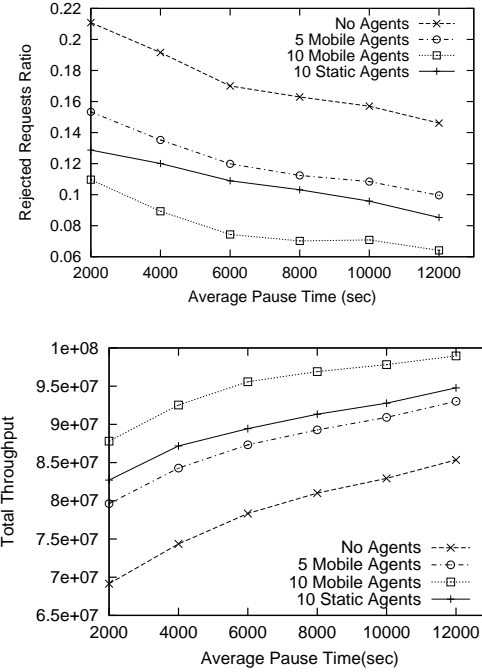


Fig. 3. Difference between using static agents and mobile agents

REFERENCES

- [1] P. Gupta and P. R. Kumar, “The Capacity of Wireless Networks,” *IEEE Transactions on Information Theory*, vol. IT-46, no. 2, pp. 388–404, March 2000.
- [2] K. Xu, X. Hong, and M. Gerla, “An Ad Hoc Network with Mobile Backbones,” in *IEEE International Conference on Communications (ICC’02)*, New York, NY, April 2002.
- [3] S. Banerjee and S. Khuller, “A Clustering Scheme for Hierarchical Control in Multi-hop Wireless Networks,” in *INFOCOM*, April 2001.
- [4] N. Riza, Ed., *Optics in Information Systems*. Special Issue: Optical Wireless Communications, Oct. 2001, vol. 12, no. 2.
- [5] R. Guerin, D. Williams, and A. Orda, “Qos routing mechanisms and ospf extensions,” in *Globecom*, 1997.
- [6] K. Kar, M. Kodialam, and T. V. Lakshman, “Minimum Interference Routing of Wavelength Guaranteed Tunnels with MPLS Traffic Engineering Applications,” *IEEE Journal on Selected Areas in Communications*, vol. 18, no. 12, pp. 2566–2579, December 2000.
- [7] M. Kodialam and T. V. Lakshman, “Integrated Dynamic IP and Wavelength Routing in IP over WDM Networks,” *INFOCOM*, April 2001.
- [8] I. Iliadis and D. Bauer, “A New Class of Online Minimum-Interference Routing Algorithms,” *Networking 2002*, vol. LNCS 2345, pp. 959–971, May 2002.
- [9] F. Sun and M. Shayman, “Minimum Interference Algorithm for Integrated Topology Control and Routing in Wireless Optical Backbone Networks,” in *IEEE International Conference on Communications*, Paris, France, June 2004.
- [10] F. Sun and M. A. Shayman, “Local restoration for bandwidth guaranteed connections in mobile optical backbone networks,” in *WCNC 2005*, March, 2005.
- [11] D. B. Johnson and D. A. Maltz, “Dynamic Source Routing in Ad Hoc Wireless Networks, Mobile Computing,” In *Mobile Computing*, edited by Tomasz Imielinski and Hank Korth, chapter 5, pages 153–181, Kluwer Academic Publishers, 1996.
- [12] R.K.Ahuja, T.L.Magnanti, and J. Orlin, *Network Flows: Theory, Algorithms, and Applications*. Prentice Hall, 1993.
- [13] G. Lin and G. Xue, “Steiner tree problem with minimum number of steiner points and bounded edge-length,” *Information Processing Letters*, vol. 67, pp. 53–57, 1999.
- [14] J. Yoon, M. Liu, and B. Noble, “Random waypoint considered harmful,” in *INFOCOM*, San Francisco, CA, 2003, pp. 1312–1321.