A Markov Chain Model for Local Path Protection in Mobile Optical Backbone Networks

Mehdi Kalantari, Fangting Sun, and Mark Shayman
Department of Electrical and Computer Engineering
University of Maryland, College Park, MD 20742
Emails: { mehkalan, ftsun, shayman }@eng.umd.edu

***Abstract*— In this paper, we propose a protection scheme to improve quality of service in a mobile backbone network with free space optical point-to-point links. We assume that the backbone nodes are mobile and with unpredictable movements, and the traffic from an ingress node is routed to its corresponding egress nodes in a multi-hop way. The QoS demanding applications are vulnerable to mobility imposed link failures in such networks. The first contribution of this work is to introduce a scheme in which a set of controllable protection agents are deployed to provide protection of the paths in the critical points of the network. The second contribution is using a Markov chain model to evaluate potential risk or potential reward of placing a protection agent in a specific place of the network. Our simulation experiments show that by using our scheme to strategically place a limited number of protection agents, a significant improvement in network performance is achieved.**

I. Introduction

A mobile ad hoc network (MANET) is composed of a group of mobile nodes without centralized administration or fixed network infrastructure. In such networks mobile nodes can communicate with other nodes directly or through cooperative forwarding nodes. It has been shown that a flat ad hoc network has poor scalability [1] in terms of per node throughput. To build a MANET with scalable throughput, a promising solution is to organize nodes in a hierarchical manner [2], [3], in which some nodes are selected to form a higher level network called the backbone network that can establish links among themselves.

In this work, 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, our results are extendable to the case in which nodes can establish directional radio frequency (RF) links with narrow beams. Furthermore, we assume bandwidth guaranteed connections (e.g., MPLS label switched paths) are created to carry aggregate traffic between source-destination pairs of the backbone network. This is reasonable since we assume limited mobility for the backbone nodes.

Due to their connection-oriented nature, bandwidth guaranteed connections are prone to failures caused by breakdown of links or movement of nodes. In most of the suggested approaches for protection and restoration of bandwidth guaranteed paths in wireline networks, the primary focus is on reserving a backup path that is disjoint from the active path. Methods in [4], [5] suggest an end-to-end backup path reservation, while the proposed approaches in [6], [7] are more focused on local backup paths. In all of the suggested approaches the backup path is used for restoring the connectivity when the active path fails. However, in mobile ad hoc networks most of the link failures are due to mobility of nodes, and most of the time failure of an active path and its backup path are correlated. The other disadvantage of backup paths is the fact that reserving two paths is not bandwidth-efficient.

Another way to alleviate the effects of connection failures is to repair a broken connection upon path failure, either locally or end-to-end. The main disadvantage of this method is the fact that it is reactive, and it incurs an intolerable delay and interruption of the ongoing communication in the broken path, and it is not suitable for most of the high QoS demanding applications.

We call an *interruption* the event in which a connection is broken and it is not immediately restored in a real time way. For bandwidth guaranteed connections, an important QoS measure is the average rate of interruptions, and it is desired to make this quantity as small as possible. There are two ways to restore the connections broken by path failures: the end-to-end restoration, in which a new path is discovered and set up between the ingress node and the egress node, and the local restoration, in which a new sub-route is discovered and set up to connect the nodes at the sides of broken links. In general, both methods need to introduce extra delay to discover and set up the new path upon path breakage. The end-to-end restoration usually introduces more delay than the local restoration, while the local restoration may generate a sub-optimal path.

In this paper, we introduce a new type of mobile node, *controllable protection agent*, to restore the broken connections in real time, and to reduce the rate of interruptions. The protection agents have the same transmission capability as the general backbone nodes. The difference lies in two aspects. First, the agents are not used to take traffic under normal circumstances, and they are only used to temporarily reroute traffic upon path failures. Second, the movement of the agents is controllable, and they can be proactively placed and dynamically relocated to protect high risk connections in the network.

We consider the following scenario: there is a set of general mobile backbone nodes and agents, which are geographically distributed in an area. The backbone nodes may be the clusterheads, which move when their clusters relocate. Each backbone node collects the traffic demands within its cluster

This research was partially supported by AFOSR under grant F496200210217.

and issues the requests for aggregate bandwidth to other backbone nodes. For each request, if sufficient resources are available, a bandwidth guaranteed connection is established between the source and destination; otherwise, the request is rejected. Since an established connection may break due to path failures, restoration mechanisms should be applied to guarantee QoS. We use the following restoration scheme in this paper: when a path is broken due to movement of the nodes contributing to the path, we first check whether there exists an agent which can temporarily take the traffic passing through the broken path. If the broken path can be repaired by a protection agent, then the traffic on the broken link is immediately rerouted through that agent, and no interruption happens; otherwise, an end-to-end path should be discovered and a new bandwidth guaranteed connection is to be established, which is counted as an interruption and degradation of the QoS.

In order to reduce the number of interruptions, it is desired to place protection agents in an optimal way and periodically update their location to adaptively follow changes in the network. Due to the limited number of protection agents, their optimal placement is a critical issue to achieve a high improvement in the network performance. We divide the network into small area elements, and for each element we define a *state*. The state of an element represents the number of calls that are routed through the nodes in that element. We define a set of discrete states for each element, and we assume the state of each element satisfies the *Markov Property*. Then we define a discrete Markov chain for an element; the state of this Markov chain is defined as the discrete time samples of the state of the element taken periodically at the end of some predefined time intervals.

The state transition dynamics of each Markov chain depends on the node mobility an traffic patterns. In order to discover these dynamics for the above explained Markov chain in an element, we use the observations of state of that element during a sliding window in the past. If the number of observations is large enough, then an accurate estimate of transition probabilities can be achieved. The last step of our proposed method is to define a stochastic cost function that represents the cost associated with an element when it has no protection agent in it. We will write the above stochastic cost function of an element in terms of the dynamics and state transition probabilities of the Markov chain of that element. Finally, we put the protection agents in the elements with the highest expected costs, and as mentioned before, we continually do the evaluation of cost for all elements and relocate the protection agents accordingly.

II. System Model and Basic Assumptions

In this section we specify our different assumptions about mobility of nodes, call arrival process, and protection mechanism. We assume there are two types of nodes in a wireless optical backbone network: general mobile backbone nodes and controllable protection agents. Each node is equipped with a number of FSO transmitters and receivers. A unidirectional FSO link can be set up between a pair of FSO transmitter and receiver within the transmission range of each other. We say a *potential link* exists from node A to node B if they are in each other's transmission range, node A has a free transmitter, and node B has a free receiver. Furthermore, we say an *actual link* exists from node A to node B if an FSO link has been established from A to B. Each transmitter or receiver can only participate in one actual link. Physical establishment of an actual FSO link between two nodes requires the procedures of pointing, acquisition and tracking. These procedures are outside the scope of this paper, and an interested reader may see [8], [9].

In the following subsections we will describe different assumptions of our model.

A. General Assumptions on Nodes

We assume there are N wireless nodes distributed in a 2-dimensional plane denoted by A. The nodes are mobile and they can freely move in A. We assume a *random waypoint* mobility model, in which a node is stationary for an exponentially distributed random pause time with mean τ_1, and then it moves to a random destination in the network area with a random speed and stays there for another exponentially distributed random pause time, and repeats this procedure infinitely.

These above nodes in the network are accompanied by M protection agents for which we have full control on their mobility. We can think of the protection nodes as Unmanned Aerial Vehicles (UAVs), or any other kind of nodes with controlled mobility that are used for protection purposes.

B. Request Arrival and Departure Process

We assume the traffic arrival process in each node is a Poisson process with rate λ. The protection nodes do not generate or terminate any traffic, and they only contribute in forwarding of the traffic of the other nodes. Furthermore, we assume the destination of a generated request is randomly selected in the network and its holding time is an exponentially distributed random time with mean τ_2. When a request is generated at one of the nodes, if a path with sufficient bandwidth exists between the source and the destination of that request, a bandwidth guaranteed connection is established; otherwise, the request is rejected.

C. Creation and Deletion of Bandwidth Guaranteed Connections

Creating a bandwidth guaranteed connection for a request consists of two steps. In the first step, the source of the request performs route discovery to find a valid route to the destination with enough resources, Such a route can be composed of actual and potential links with the following conditions: (1) for each actual link, the available bandwidth is at least the amount of bandwidth requested; (2) for each potential link 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 ad hoc routing protocols, such as DSR [10]. After the successful route discovery, all potential links along the path are established as actual links.

A reserved path and it corresponding resources are released after termination of its corresponding request. Another condition on which a reservation on a link is released is the situation under which a communication is interrupted as a result of a node mobility. In this condition the path that passes through that node is broken and it is not repaired due to unavailability of protection agents. In order to free up the transmitters and receivers of the nodes, links with zero reservation or zero utilization are torn down.

D. Protection Mechanism

Our objective is to place the protection agents in proper places in the network to provide a proactive mechanism to protect the ongoing communication and to provide backup for critical nodes. When a node contributing to a bandwidth guaranteed connection moves away from its current position it causes a path failure; if a protection agent is available at the place of path failure, it will immediately repair the path without any interruption, otherwise an interruption happens. We consider the average number of interruptions as the performance measure of our protection schemes.

An important fact is that protection agents locally repair the paths, and in order to make the maximum use of them, after a local repair is performed by them, the source corresponding to the request performs the signalling process explained in the previous subsection to discover a new end-to-end path to its destination. As soon as discovery of such a path, the traffic is switched to that path and the protection agent is released.

III. Placement of Protection Agents

In this section, we suggest a scheme for proper placement of protection agents in the network. For this purpose, we present a Markov Chain structure that models the dynamic of communication load in the different geographical areas of the network. We partition the network into small pieces and define a Markov chain the models the dynamics of the level of communication activity in each piece. Then we use the Markov chain of each piece to evaluate the expected cost of not placing a protection agent in that piece.

Assume the network area A is partitioned into K disjoint connected subsets. We call each subset an element. Let $B_i(t)$ denote the set of all nodes that reside inside the i^{th} element at time t. Also assume $f_n(t)$ represents the amount of traffic forwarded by node n at time t. Note that $f_n(t)$ does not include the traffic originated at n or the traffic destined to node n. We define the state of the i^{th} element as:

$$\zeta_i(t) = \sum_{n \in B_i(t)} f_n(t) \tag{1}$$

In other words, $f_n(t)$ is the total traffic that is being routed through the i^{th} element at time t.

It is important to mention that the validity and accuracy of using a Markov Process depends on the validity of the Markov property for the above defined state:

$$P\{\zeta_i(t_1) = z_1 | \zeta_i(t_0) = z_0,\ \zeta_i(t) = z(t)\ \ -\infty \leq t \leq t_0\} = \\ P\{\zeta_i(t_1) = z_1 | \zeta_i(t_0) = z_0\} \tag{2}$$

in which $t_0 \leq t_1$. The Markov property simply states among all the information of the state from the past, the most recent one is a sufficient statistic for estimation of the state in the future. Although the above property is hard to prove in our case, under the assumptions of the previous section, request durations and pause times of mobile nodes are both exponential random variables, and based on the memoryless nature of the exponential random variable, we expect that the Markov assumption gives a good approximation of the dynamics of the communication activity in an element.

In the next step, we define a discrete time version of the above Markov process for the i^{th} element. For this purpose, we discretize both time and the the value of the state. Starting at $t = 0$, we define $S_i(k)$ as the state of this Markov chain at the k^{th} step. $S_i(k)$ can be written in terms of the state of the continuous Markov process as follows:

$$S_i(k) = \lfloor \zeta_i(kT_s)/D \rfloor \tag{3}$$

where T_s is the discretization sampling time interval, and D is the interval for discretization of the state value. In (3), $\lfloor x \rfloor$ represents the greatest integer value not larger than x. The state of the above Markov chain takes nonnegative integer values.

In order to complete Markov chain model, we need the transition probabilities between each pair of states. Our solution for acquisition of the transition probabilities is to estimate them based on the numerical value of the state in a large enough time interval over which enough observation of the value of state has been collected. For example, if on a window of length L we have L sequential observations of the state of the Markov chain of the i^{th} element, then single step transition probability from state s_1 to state s_2 can be written as:

$$p_i(s_1, s_2) = \frac{\sum_{j=1}^{L-1} 1(S(j) = s_1 \text{ and } S(j+1) = s_2)}{\sum_{j=1}^{L-1} 1(S(j) = s_1)} \tag{4}$$

in which $1(x)$ is the indicator function, and it returns 1 if statement x is true, and it returns 0 otherwise.

From a practical point of view, in the Markov chain defined by equation (3), we do not need infinite number of values for the state, and the state can be truncated at a high enough integer value, for which there is a low probability of occurrence under the steady state conditions. We call this value S_{max}. Therefore, we can rewrite the definition of state in the following way:

$$S_i(k) = min(\lfloor \zeta_i(kT_s)/D \rfloor, S_{max}). \tag{5}$$

With definition $S_i(k)$ can take only $S_{max} + 1$ integer values.

Now we are ready to use our Markov Chain model to define a measure of cost in different elements of the network. For an element, we define per step cost $C(s)$ as the one step cost of being in state s. $C(s)$ has an abstract definition, and we can think of it as the one step cost or its corresponding risk when an element with state value of s is left unprotected. $C(s)$ is a nondecreasing and nonnegative function of s, and we can define $C(0) = 0$. In other words, if there is no communication or no node in an element, we do not have to pay any cost for not protecting that element.

Assume $\{S_i(k)\}_{k=0}^{\infty}$ takes its values according to a given sequence $\{s_0, s_1, s_2, ...\}$. In other words, $S_i(k) = s_k$. Then we define the following discounted *cost to go* for the i^{th} element:

$$J_i = \sum_{k=0}^{\infty} \gamma^k C(s_k) \tag{6}$$

in which $0 < \gamma < 1$ is a discount factor. For the values of γ closer to one, we give a higher weight to the cost per step of states in the far future, and for a γ close to zero, we give a higher weight to the values of the cost in near future. If we set the current time as the reference ($k = 0$), then the only deterministic value in $\{s_0, s_1, s_2, ...\}$ will be s_0. Then the value of cost to go in equation (6) will be a random variable. The expected value of this random variable will be a function of s_0, and it can be defined in the following way:

$$\bar{J}_i(s) = E\{J_i | s_0 = s\} \tag{7}$$

It is straightforward to verify that:

$$\bar{J}_i(s) = C(s) + \gamma \sum_{s'=0}^{S_{max}} p(s, s')\bar{J}_i(s') \tag{8}$$

The proof of equation (8) can be found in [11]. If we write equation (8) for all of the states, we will get $S_{max} + 1$ linear equations with $S_{max} + 1$ unknowns. This system of equations can be written in the matrix from as:

$$(I - \gamma P)\underline{J} = \underline{C} \tag{9}$$

in which, $\underline{J} = [\bar{J}_i(0)\ \bar{J}_i(1)\ ...\ \bar{J}_i(S_{max})]^T$, $\underline{C} = [C(0)\ C(1)\ ...\ C(S_{max})]^T$, P is the $(S_{max}+1) \times (S_{max}+1)$ matrix that has the transition probabilities as its entries. In other words: $P_{ss'} = p(s, s')$. Finally, I in equation (9) is the $(S_{max} + 1) \times (S_{max} + 1)$ identity matrix. It can be shown that under the assumption of $0 < \gamma < 1$ the matrix $I - \gamma P$ is nonsingular, so the system of equations defined by equation (9), is always solvable.

The final step to complete our algorithm is to place protection agents in the elements with the highest cost to go. Assuming the number of protection agents, M, to be considerably smaller then the number of elements K, we calculate the cost to go for all of the elements, and sort them in decreasing order. Then the protection agents are assigned to the first M elements in the list. This assignment is dynamic and the placement of the protection agents is updated periodically in the network.

The nodes inside every element are in charge of observing the state, and performing all above calculations. All the nodes in an element share the information of the observed state over the window period for which the transition probabilities are estimated. If a node leaves and element, its observation information is reset, and information of the element that it has newly arrived at is acquired from the nodes residing in it. If a node arrives at an element with no node, it will try to collect the observation information from scratch. The nodes of each element broadcast or flood their calculated cost to go in a periodic basis.

A very interesting note about the above Markov chain model is the fact that all dynamical complexities of the system that result from the request arrival process and mobility of the nodes are summarized in the the transition probabilities of the Markov chain, and these transition probabilities are identified by observation of the state over a window period.

Proper definition of the elements is a sensitive issue in our approach. Making the elements too small causes the network to be divided into too many elements, and subsequently, it may happen that most of the elements are empty. On the other hand, making the elements too large does not give accurate and specific information about placement of agents. The diameter of a connected set in the plane is defined as the supremum of the Euclidian distance of two arbitrary points of that set. With this definition, our intuition is that elements with diameter about half the transmission range of nodes give a good performance. This is because the fact that by placing a protection node inside an element, it should be able to to repair paths passing through that element by being able to connect to the nodes in the neighboring elements.

One of the easiest ways of defining elements is to divide the network area through vertical and horizontal grids. Such grids partition the network area into small rectangles, and each rectangle is defined as an element.

IV. Performance Studies

A. Simulation Methodology

We use a rectangular space of size 10000m × 10000m. The total number of general 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. There are 40 source-destination traffic pairs randomly generated for each simulation. For each traffic pair, the bandwidth guaranteed connection request arrival time is modelled as a Poisson process, and the average request interarrival time is chosen uniformly between 500 to 1000 sec. For each request, the duration is modelled as an exponentially distributed random variable with mean 2500 sec, and the bandwidth demand is uniformly distributed between 2 to 6 units. Upon arrival of a request, Dynamic Source Routing (DSR) [10] is used for route discovery.

In the simulations, each backbone node moves randomly according to a *random waypoint model* [10]: A node starts at a random position in the network, waits for a duration called the *pause time*, which is modelled as a random variable with exponential distribution, 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. In the simulations, we set v_{min} to be 10m/s, v_{max} to be 20m/s, and use different average pause time.

The whole network is partitioned into 100 same size elements. Each element maintains its own Markov chain that has 5 discrete states, and for each state we use per step cost $C(i) = i$. We set sampling interval T_s as 15 sec, the observation window $L = 200$, the interval for discretization of the state $D = 15$ units and the discount factor $\gamma = 0.8$. All experiments are run for 20000 seconds.

B. Simulation Evaluation

We first examine the average interruption number under different situations. Fig. 1 shows the simulation results for the average number of interruptions per request under various number of protection agents and different average pause time. From Fig. 1 it can be observed that the average number of interruptions per request decreases quickly with the increase in the number of protection agents, especially when the number of agents is not too large. For example, when we increase the number of protection agents from 0 to 5, the number of interruptions per request is decreased by 32.1%, 34.2% and 31.4% when average pause time is 5000, 7500 and 10000 seconds respectively.

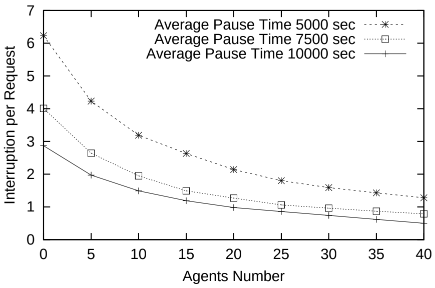

Fig. 1. Number of interruptions per request using different number of agent

By inspecting Fig. 1 we can also see that the requests are interrupted much less frequently when some agents are added into the network compared with no agents. For example, for the average pause time of 5000 seconds, by adding 15 agents to the network, the request is interrupted every 16 minutes on the average; while the request is interrupted every 6.5 minutes on the average without agents.

From Fig. 1 it is also easy to see that there is little benefit from additional agents once the number of agents reaches 20, and the interruption may not go to zero. One way to explain this phenomenon is that our current approach only permits two-hop repair of the broken paths by using protection agents, but some of the possible path failures require more than two hops to repair them.

Fig. 2 shows the comparison results under different protection agent setup. In Fig. 2, “mobile agents” represents the experiment in which protection agents can be dynamically relocated, while “static agents” represents the experiment in which protection agents are randomly distributed inside the area according to a uniform distribution, and they are not relocated during the whole simulation time. From Fig. 2 we can see that the improvement achieved by 10 mobile agents is significantly better than the improvement achieved by 10 static agents. It can even be seen that 5 mobile agents give a better performance than 10 static agents.

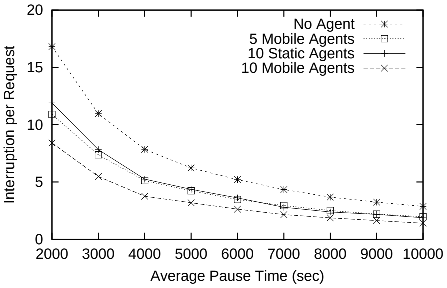

Fig. 2. Difference between using static agents and mobile agents

V. Conclusion

In this work we proposed a new type of controllable mobile node for protecting the bandwidth guaranteed connections in the backbone of an ad hoc network with point-to-point wireless optical links. Furthermore, we proposed a scheme for local restoration of broken paths failed by the mobility of nodes. In our scheme, we partition the network area to small connected and disjoint elements, and for each element, we define a Markov chain model to evaluate the expected cost in that element when no protection is provided. Also we suggested an algorithm that places the protection agents in the elements with the highest cost. Our simulation experiments show that the proposed scheme achieves a significant improvement in the performance of the network.

References

[1] P. Gupta and P. R. Kumar, “The Capacity of Wireless Networks,” *IEEE Transactions on Information Theory*, vol. IT-46, no. 2, March 2000.

[2] S. Banerjee and S. Khuller, “A Clustering Scheme for Hierarchical Control in Multi-hop Wireless Networks,” in *IEEE Infocom 2001.*

[3] K. Xu, X. Hong, and M. Gerla, “An Ad Hoc Network with Mobile Backbones,” in *IEEE International Conference on Communications (ICC’02).*

[4] M. S. Kodialam and T. V. Lakshman, “Dynamic Routing of Bandwidth Guaranteed Tunnels with Restoration,” in *IEEE Infocom 2000.*

[5] K. Kar, M. Kodialam, and T. V. Lakshman, “Routing Restorable Bandwidth Guaranteed Connections using Maximum 2-Route Flows,” in *IEEE Infocom 2002.*

[6] M. S. Kodialam and T. V. Lakshman, “Dynamic Routing of Locally Restorable Bandwidth Guaranteed Tunnels Using Aggregated Link Usage Information,” in *IEEE Infocom 2001.*

[7] L. Mélon, F. Blanchy, and G. Leduc, “Decentralized Local Backup LSP Calculation with Efficient Bandwidth Sharing,” in *IEEE International Conference on Telecommunications (ICT 2003).*

[8] N. Riza, *Optics in Information Systems*, Special Issue: Optical Wireless Communications, 2001.

[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 (ICC 2004).*

[10] 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.

[11] D. Bertsekas, *Dynamic Programming and Optimal Control, 2nd Edition*, Athena Scientific, 2001.