Lazy Flooding: A New Technique for Signaling in All Optical Network

C.-X. Chi, D.-W. Huang, David Lee and X.-R. Sun
chic,dhuang4,leedavid,xsun1@lucent.com; Bell Laboratories

Abstract: We propose a new signaling technique for resource and topology information dissemination in all optical networks: lazy flooding. It significantly cuts down the number of floods and reduces the signaling network congestion.

1. Introduction

In a distributed all optical network, resource and network topology information has to be constantly updated at each optical switch, such that correct lightpath decision can be made in a dynamic and distributed networking environment. In an all optical network such as Lucent Lambda Router Network [2], the topology information, which includes the Optical Cross Connect (OXC) up and down and the fiber (link) cut and repair, is flooded throughout a separate Data Communication Network (DCN), which is a signaling network. In addition, when a lightpath is set up and torn down all the involved channel status is changed from being available to occupied and vice versa, and this information is also flooded throughout the DCN network. Obviously, for a network with N nodes (OXC), each channel, link or node status change results in an order of N^2 messages to be flooded via the DCN [1], leading to signaling network congestion and instability, also making it difficult to scale up.

To cope with this optical networks DCN congestion problem we propose a new technique: lazy flooding. It significantly reduces the number of flooded messages for the network resource and topology information update without affecting the lightpath construction and, consequently, path protection and restoration. The idea is simple: a channel status is flooded only when the number of available channels is from a selected set of values. Apparently, this method is "lazy", i.e., it does not flood for each channel update, and we call it *lazy flooding.* Different selected sets of channel numbers for flooding give different lazy flooding methods; a message is flooded whenever the number of the available channels on a link is: (1) Below a threshold (Threshold Flooding); (2) Following a geometric sequence (Exponential Flooding); and (3) Following a Fibonacci sequence (Fibonacci Flooding).

We shall show that these lazy flooding techniques significantly reduce the number of messages flooded throughout the DCN. On the other hand, lazy flooding might cause discrepancy in link load information and unnecessary link and network blocking. Mathematical analysis and simulation show that these effects from lazy flooding are negligible.

2. Lazy Flooding Methods and Mathematical Analysis

We consider a discrete time and discrete state link (fiber) with a total B channels: the link capacity. Assume that the available number of channels $\{c_t, t = 0, 1, 2, ...\}$ follows an Ergodic Markov Chain with a transition probability $\Pr(c_{t+1} = j | c_t = i) = p_{ij}$. Let $\rho = \frac{\lambda}{\mu}$, From [3], the stationary distribution is: $\pi_k = \frac{\rho^{B-k}}{(B-k)!}\pi_B \; \pi_B = \frac{1}{\sum_{k=0}^{n} \frac{\rho^k}{k!}}$; $k = 0, ..., B-1$.

A straightforward flooding method, which is used in Internet OSPF, is **All Flooding**: the link capacity is flooded whenever there is a change. It has the flooding probability $p_{ij} = 1$ for $|i-j| = 1$, $i, j = 0, 1, \cdots, B$. Lazy flooding is different; it only floods when the link capacity $c_t \leq B$ reaches a specified value and has a change. Different sets of specified values of link capacities for flooding give different lazy flooding methods:

(1) **Threshold Flooding** It floods if and only if the link capacity is below a threshold value L, i.e., $c_t \leq L$ and makes a change. The flooding probability is

$$P_t = \Pr[c_{t+1} \neq c_t, c_t \leq L] = B\mu\pi_0 + \sum_{k=1}^{L} [(B-k)\mu + \lambda]\pi_k.$$

(2)**Exponential Flooding**: Let $b_k = \begin{cases} k, & k = 0, 1, ..., L; \\ L + 2^{k-L}, & k = L+1, ...K \end{cases}$ where $K = \lfloor \log_2(B - L) \rfloor + L$. Note that b_k follows a geometric sequence above the threshold L. It floods if and only if the link capacity changes from the current $c_t = b_k$. The flooding probability is: $P_e = \Pr\{\cup_k [c_t = b_k, c_{t+1} \neq b_k]\} = B\mu\pi_0 + \sum_{k=1}^{K} [\lambda + \mu(B - b_k)] \pi_{b_k}$.

(3)**Fibonacci Flooding**: Let $f_k = \begin{cases} k, & k = 0, 1, ..., L; \\ f_{k-1} + f_{k-2} - L + 3, & k = L+1, ...K \end{cases}$ where K is the largest index k such that $f_k \leq B$. Note that f_k follows a Fibonacci sequence above the threshold L. It floods if and only if the link capacity changes from the current $c_t = f_k$. The flooding probability is:

$$P_f = \Pr\{\cup_k [c_t = f_k, c_{t+1} \neq f_k]\} = B\mu\pi_0 + \sum_{k=1}^{K} [\lambda + \mu(B - f_k + 1)] \pi_{f_k}.$$

Intuitively, All Flooding requires B floods, Threshold Flooding requires L flood, and Exponential and Fibonacci Flooding require approximately $L + log\ B$ floods. Figure 1(a) shows the flooding probabilities of the three flooding polices over different model parameter $\rho = \frac{\lambda}{\mu}$ and threshold $L = 5$. Overall the three lazy flooding methods have a similar flood probability and are much smaller than that of All Flooding.

On the other hand, due to lazy flooding other nodes in the network do not know the exact capacities of links. To measure the missing information, we compute the mean and variance of the difference between the exact link capacity c_t and the capacity $\hat{c}_t$ flooded to the network. For Threshold Flooding, the mean and variance are:

$$E(c_t - \hat{c}_t) = \sum_{k=L+1}^{B} (k - L)\pi_k, V(c_t - \hat{c}_t) = \sum_{k=L+1}^{B} [k - L - E(c_t)]^2 \pi_k + \sum_{k=0}^{L} [E(c_t)]^2 \pi_k.$$

For Exponential (Fibonacci, replace b_k by f_k) Flooding:

$$E(c_t - \hat{c}_t) = \sum_{k=L}^{K} \sum_{j=b_{k-1}+1}^{b_{k+1}-1} (j - b_k) p_{k,j}, \quad V(c_t - \hat{c}_t) = \sum_{k=L}^{K} \sum_{j=b_{k-1}+1}^{b_{k+1}-1} [j - b_k - E(c_t - \hat{c}_t)]^2 p_{k,j}.$$

where $p_{k,j} = \left(1 + \sum_{l=1}^{j-b_{k-1}-1} \prod_{m=1}^{l} \frac{B-j+m}{\rho}\right) p_{k,b_{k-1}+1}$

We plot the mean and variance of the three lazy flooding methods with threshold $L = 5$ against different $M/M/B$ model parameter ρ in Figure 1(b)(c). It is obvious that Threshold Flooding exhibits large discrepancy in mean of the differences since it does not flood at all when the capacity is above the threshold. On the other hand, Exponential and Fibonacci Flooding has rather small mean and variance in the discrepancy especially for large ρ values; they disseminate almost the same network topology and resource information as All Flooding but with much less floods.

3. Simulation Study for Flooding Schemes

For the overall network performance, we conduct a simulation study. We compare the link and network blocking rates when all-flooding and lazy flooding schemes are used. We shall conclude that lazy flooding and all flooding have more or less the same link and network blocking rate while lazy flooding has much less flooded messages in the DCN signaling network. We experiment on two routing algorithms: **Algorithm 1** Link weight is insensitive to the load: every link is assigned a fixed $weight = 1$; and**Algorithm 2** Link weight is sensitive to the load: every link is assigned $weight = (2 \times H + 1)^\alpha$ ($\alpha = 1 - \frac{C_t}{B}$ is the load of the link and H is the network diameter) where the link weight increases exponentially with its traffic load. For the assigned link weight, we construct lightpaths using Dijkstra's shortest path algorithm. The simulation is based on a given 14 nodes network which has a similiar topology as NSFnet in U.S.A. Each link is bi-directional and has a capacity of $B = 20$ channels. We experiment on different network load. We study: (1) $Link\_block\_rate = \frac{\sum_{i=1}^{M} R_i}{\sum_{i=1}^{M} C_i}$, where M is the total number of links in the network, R_i is

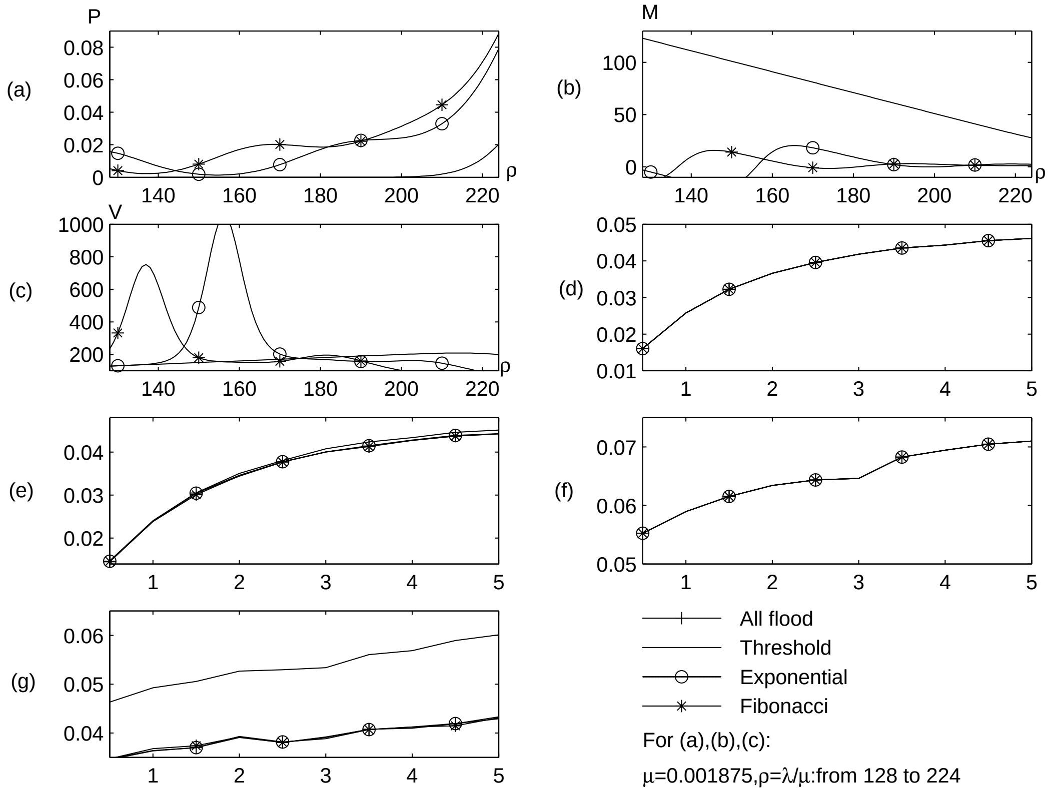

Figure 1: Numerical Results for Lazy Flooding Schemes

the number of connections which are rejected because link i has no channels, and C_i is the total capacity of link i; and (2) $Network\_block\_rate = \frac{R}{N}$, where R is the number of connections that are rejected, and N is the number of connection requests to the network.

Figure 1(d)(e) show the mean of the link blocking rate, and Figure 1(f)(g) show the network blocking rate. Their x-axis is the average network flooding delay. We conclude: (1) When the traffic load insensitive algorithm 1 is used, the link blocking rate and network blocking rate is the same no matter what flooding scheme is used to advertise the network status. Lazy flooding schemes have much less floods than all-flooding but with the same link and network blocking rate. (2) When the traffic load sensitive algorithm 2 is used, in general the more accurate the network status is available, the less link blocking rate and network blocking rate are. Exponential and Fibonacci flooding have much less floods than all-flooding but with approximately the same link blocking rate. With slightly more floods Exponential and Fibonacci Flooding have much lower network blocking rate than Threshold Flooding. On the other hand, comparing with all flooding, they have much less floods but almost the same network blocking rate.

In summary, with much less flooding messages, lazy flooding achieves almost the same performance as all-flooding, yet reduces the DCN congestion. Among the three lazy flooding techniques, Exponential and Fibonacci Flooding has slightly more floods but is superior than Threshold Flooding.

reference

[1] A.Aho,and D.Lee, "Hierarchical Networks and the LSA N-Squared Problem in OSPF Routing", Proc. Globecom 2000.

[2] "Optical Network Management Protocol", Bell Labs Tech Memo. August 2001.

[3]L. Kleinrock (1975), Queueing systems, Volume 1: Theory, p105-106, John Wiley & Sons.