Back Pressure-Based Adaptive Routing Algorithm Technique in Communication Networks

D. Anosh Hussain[1], V. Padmaja[2]

[1] M.Tech (CS), Nimra College of Engineering and Technology, A.P., India.

[2]Asst. Professor, Dept. of Computer Science & Engineering, Nimra College of Engineering and Technology, A.P., India.

Abstract— Routing Packets effectively and efficiently is the major task in networking. The basic idea of backpressure techniques is to prioritize transmissions over links that have the highest queue differentials. Backpressure method effectively makes packets flow through the network as though pulled by gravity towards the destination end, which has the smallest queue size of zero. Under high traffic conditions, this method works very well, and backpressure is able to fully utilize the available network resources in a highly dynamic fashion. Under low traffic conditions, however, because many other hosts may also have a small or zero queue size, there is inefficiency in terms of an increase in delay, as packets may loop or take a long time to make their way to the destination end. In this paper we use the concept of shadow queues. Each node has to maintain some counters, called as shadow queues, per destination. This is very similar to the idea of maintaining a routing table (for routing purpose) per destination. Using the concept of shadow queues, we partially decouple routing and the scheduling. A shadow network is maintained to update a probabilistic routing table that packets use upon arrival at a node. The same shadow network, with back-pressure technique, is used to activate transmissions between nodes. The routing algorithm is designed to minimize the average number of hops used by the packets in the network. This idea, along with the scheduling and routing decoupling, leads to delay reduction compared with the traditional back-pressure algorithm.

Keywords— Back pressure, Packets, Routing, Shadow queue.

I. INTRODUCTION

The back pressure method first proposed in [1] has recently displayed a great potential for solving a number of issues in the wireless multi-hop networks. The main idea of back-pressure scheduling model is that contention among the links should be resolved by scheduling the link which has the largest product of the queue differential backlog between its endpoints and the transmission rate at which the link can be served. In a perfectly time-slotted medium access mechanism such as TDMA (Time Division Multiple Access), this will result into optimal throughput of flows while guaranteeing the queue stability (ingress traffic to a queue never exceeds its egress traffic). The utility maximization framework initially proposed in [2] shows that injection rates of the flows should be chosen such that aggregate utility of the flows is to be maximized. Here the utility of flow represents a desirable effect on the entire network achieved by a particular rate of the flow. It was shown in [3] [4] that the backpressure scheduling and utility based rate control together can solve the global problem of the network utility maximization.

The fundamental challenge with back pressure method is that solution of the underlying scheduling strategy is NP-hard [5]. Also, since it was proposed for a centralized, time slotted, and synchronized system, a distributed implementation which can achieve even a closer approximation is very difficult to develop. Recently, [6] have attempted to incorporate backpressure technique based scheduling in random medium access protocols such as CSMA/CA. These protocols try to approximate the performance of the ideal back pressure scheduler by prioritizing the frame transmissions according to differential backlogs of the queues. Here, every node in the network maintains a per destination queue (PDQ) and the packets destined to a particular destination host are stored in the PDQ of that destination host until further forwarding decisions are made. Now, nodes share their PDQ information with their neighboring nodes, and this information is utilized by every node to calculate the differential backlogs of its PDQs. The differential backlog of a PDQ at a node is equal to the size of the PDQ minus the size of the PDQ of its upstream neighbor towards the destination end. To emulate the back pressure scheduling, the packets of the PDQ which has the highest differential backlog (highest back-pressure) in the neighborhood are given the higher chances for the transmission. This way, the likelihood that the packets are transmitted from a particular PDQ at a node is proportional to its differential backlog compared to the differential backlogs of PDQs of all nodes in the neighborhood. This prioritization quickly moves the traffic from long back logged queues to the shorter queues achieving an improved throughput and a better overall stability of queues.

II. RELATED WORK

The first theoretical work on backpressure scheduling technique is the classic result in [1], proving that this queue-differential based scheduling method is throughput optimal (i.e., it can stabilize any

feasible rate vector in a network). Since then, researchers have combined the basic backpressure technique with utility optimization to provide a comprehensive approach to the stochastic network optimization [7], [8]. In [9], the authors present a method whereby only one real queue is maintained for each neighbor node, along with virtual counters and shadow queues for all destinations, and show that this yields delay improvements. In [10], a novel variant of the backpressure scheduling technique is proposed which uses head of line packet delay instead of the queue lengths as the basis of the backpressure weight calculation for each link or commodity, also yielding enhanced delay performance. However, these works both assume the existence of the static fixed routes.

In [11] and [12], the authors present two works on backpressure routing aimed specifically for cluster-based intermittently connected networks. In [11], the authors develop a two phase routing method, combining the backpressure routing with source routing for the cluster-based networks, separating intra cluster routing from inter-cluster routing. They show that this scheme results in large queues at only a subset of the nodes, yielding smaller delays than conventional backpressure technique. In [12], the authors implement the above-mentioned algorithm in a real experimental network and show that the delay improvements empirically. The key difference of these works from our proposed work is that we do not make any assumption about the intermittently connected network being organized in a cluster based hierarchy and we require no previous knowledge of nodes mobility.

III. PROPOSED WORK

A. Throughput Optimal Backpressure Algorithm

This algorithm works on wireless networks. This back pressure algorithm should be depending on one main procedure. The main procedure is the destination queue procedure by using these procedure only we can reducing the poor delay of the time and the distance every node at the time of routing file in between the two or more locations predestination queues maintaining the two phases:

- routing algorithm
- scheduling algorithm.

In this system says distance is very high and time is very high. So may draw backs will be occurred at the time of working on the routing algorithm that's why we are implementing a new algorithm. Scheduling algorithm combined with the routing algorithm. Our scheduling algorithm says what to providing the schedule for prepress of the over come to draw backs that's why we are using the round robin algorithm and shortest path routing algorithm. In this algorithm providing quality of services to the end users

depending on the service only routing will be decide it is best routing or the bad routing . QOS may be depends 4 phases: Bandwidth, frequency, time, distance. In any network depends on QOS. Bandwidth and frequency is very high and time distance is low QOS is very high. At each link the algorithm assigns a weight to each possible destination node that is called back pressure. Define the back pressure at link (n,j) for destination node d at slot t to be:

$$w^d_{n,j}(t) = Q_{nd}(t) - Q_{jd}(t)$$

where Qnd(t) denotes the number of packets at node n denoted for node d at the beginning of time-slot t. Under this notation, Qnn (t) =0, $\forall$ t. Assign a weight wnj to each link (n, j), where wnj is defined to be the maximum back pressure overall possible destinations. i.e.

$$w_{nj}(t) = \max_{d} w^d_{nj}(t)$$

B. Min Resource Routing

The back-pressure algorithm explores all paths in the entire network and, as a result, may choose paths that are unnecessarily long, which may even contain some loops, thus leading to poor performance. We address this problem by introducing a cost function that measures the total amount of the resources used by all flows in the network. Specially, we add up the traffic loads on all links in the network and use this as our cost function. The goal then is to minimize this cost subject to the network capacity constraints. Given a set of packet arrival rates that lie within the capacity area, our goal is to find the routes for the flows so that we use as few resources as possible in the network. Thus, we formulate the following optimization problem:

$$\min \sum_{(nj) \in L} \mu_{nj}$$

$$s.t. \sum_{f \in F} x_f I\{b(f) = n, e(f) = d\} + \sum_{(ln) \in L} \mu^d_{ln} \leq \sum_{(nj) \in L} \mu^d_{nj}$$

$$\forall d \in N, n \in N$$

$$\{\mu_{nj}\}(n, j) \in L$$

Figure 1 illustrates how the M– back pressure algorithm works in a simple wireline network. All links in the network can be activate simultaneously without interfering with each other. Note that the backlog difference of route 1 is 6 and the backlog difference of the route 2 is 4. Because the backlog difference of the route 2 is smaller than M, the route 2

is blocked at current traffic load. The M -back-pressure algorithm will automatically choose the route 1, which is shorter. Therefore, a proper M can avoid the long routes in when the traffic is not close to capacity.
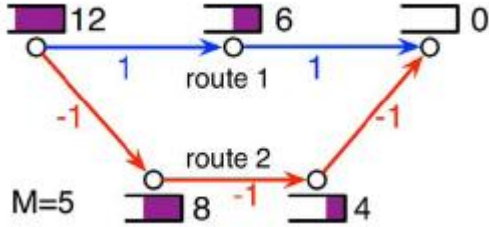

Figure 1: Link weights under the M -back-pressure algorithm.

C. Shadow Queue Algorithm

In additional to the real queues, each node n also maintains a counter, which is known as shadow queue, pnd for each destination d. Unlike the real queues, the counters are much easier to maintain even if the number of counters at each node grows linearly with the size of the overall network. The shadow queues are updated based on the movement of the fictitious entities called as shadow packets in the network. The movement of the fictitious packets can be thought of as an exchange of the control messages for the purposes of the routing and schedule. Just like the real packets, shadow packets arrive from outside the network and eventually exit the network. The back-pressure for destination on link (n j)is taken to be as follows:

$$w_{nj}^d[t] = p_{nd}[t] - p_{jd}[t] - M$$

The evolution of the shadow queue $p_{nd}[t]$ is

$$p_{nd}[t+1] = p_{nd}[t] - \sum_{j:(nj)\in L} I_{\{dn_j[t]=d\}} u_{nj}[t] +$$

$$\sum_{l:(ln)\in L} I_{\{d\ln[t]=d\}} u_{\ln}[t] + \sum_{f\in F} I_{\{b(f)=n,e(f)=d\}} a_f[t]$$

Where $u_{nj}[t]$ is the number of shadow packets transmitted over the link (n j) in time t. $d_{nj}[t]$ is the destination node that has the max weight on the link (n j). $a_f[t]$ is the number of shadow packets generated in time t for flow f.

D. Adaptive Routing Algorithm

When a packet arrived at a node, then it use routed as follows: Let $\sigma_{nj}^d[t]$ is the number of shadow packets from node n to the node j for destination d during time t. Let $\overset{\wedge}{\sigma}{}_{nj}^{d}$ denote an estimate of the expected value of $\sigma_{nj}^d[t]$ at time t. At each time slot t, the following sequence of operations occurs at each node . A packet arriving at node n for destination d is inserted in the real queue q_{nj} for next-hop neighbor j with probability:

$$P_{nj}^d[t] = \frac{\overset{\wedge}{\sigma}{}_{nj}^{d}[t]}{\sum_{k:(nk)\in L} \overset{\wedge}{\sigma}{}_{nk}^{d}[t]}$$

Packets waiting at link (n j) are transmitted over the link when that link is scheduled as shown in figure 2.
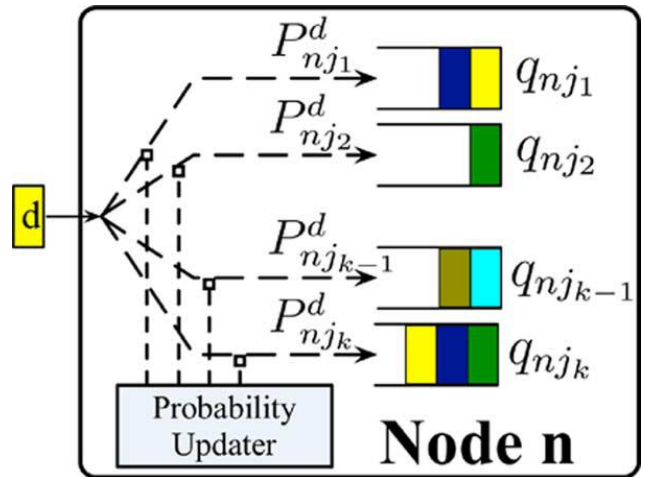

Figure 2: Probabilistic splitting algorithm at node

IV. CONCLUSION

Backpressure scheduling and routing, in which packets are preferentially transmitted over communication links with high queue differentials, offers the promise of throughput-optimal operation for a wide range of the communication networks. However, when the traffic load is very low, due to the corresponding low queue occupancy, then the backpressure scheduling/routing experiences long delays. This is particularly of concern in the intermittent encounter-based mobile networks which are already delay-limited due to the sparse and highly dynamic network connectivity. While state of the art mechanisms for such networks have proposed the use of the redundant transmissions to improve the delay,

they do not work well when the traffic load is high. We decouple (to a certain degree) routing and scheduling in the network through the use of the probabilistic routing tables and the so-called shadow queues. We use the same number of shadow queues as the back-pressure technique, but the number of real queues is very small (per neighbor). The new idea here is to perform the routing via probabilistic splitting, which allows the dramatic reduction in the number of the real queues. Finally, an important observation in this work is that the partial "decoupling" of the shadow back-pressure and real packet transmission allows us to activate more links than a regular back-pressure method would. This idea appears to be essential to reduce the delays in the routing case.

REFERENCES

[1] L. Tassiulas and A. Ephremides, "Stability properties of constrained queueing systems and scheduling policies for maximum throughput in multihop radio networks," dec 1990, pp. 2130 –2132 vol.4.

[2] F. Kelly, A. Maulloo, and D. Tan, "Rate control in communication networks: shadow prices, proportional fairness and stability," in Journal of the Operational Research Society, vol. 49, 1998.

[3] L. Chen, S. H. Low, M. Chiang, and J. C. Doyle, "Cross-layer congestion control, routing and scheduling design in ad hoc wireless networks," in INFOCOM 2006. 25th IEEE International Conference on Computer Communications. Proceedings, 2006, pp. 1 –13.

[4] A. Eryilmaz and R. Srikant, "Joint congestion control, routing, and mac for stability and fairness in wireless networks," Selected Areas in Communications, IEEE Journal on, vol. 24, no. 8, pp. 1514 –1524, 2006.

[5] M. J. N. Georgiadis Leonidas and L. Tassiulas, "Resource allocation and cross-layer control in wireless networks," in Foundations and Trends in Networking, 2006.

[6] U. Akyol, M. Andrews, P. Gupta, J. Hobby, I. Saniee, and A. Stolyar, "Joint scheduling and congestion control in mobile ad-hoc networks," april 2008, pp. 619 –627.

[7] X. Lin, N. Shroff, and R. Srikant, "A tutorial on cross-layer optimization in wireless networks," IEEE Journal on Selected Areas in Communications, vol. 24, no. 8, 2006.

[8] M. J. Neely, "Stochastic network optimization with application to communication and queueing systems," Synthesis Lectures on Communication Networks, vol. 3, no. 1, 2010.

[9] L. Bui, R. Srikant, and A. Stolyar, "Novel architectures and algorithms for delay reduction in back-pressure scheduling and routing," IEEE INFOCOM 2009.

[10] B. Ji, C. Joo, and N. Shroff, "Delay-based back-pressure scheduling in multi-hop wireless networks," IEEE INFOCOM, 2011.

[11] J. Ryu, L. Ying, and S. Shakkottai, "Back-pressure routing for intermittently connected networks," IEEE INFOCOM 2010.

[12] J. Ryu, V. Bhargava, N. Paine, and S. Shakkottai, "Back-pressure routing and rate control for icns," ACM Mobicom, 2010.