A Class of Backpressure Algorithms for Networks Embedded in Hyperbolic Space with Controllable Delay-Throughput Trade-off

Eleni Stai
Institute of Communications and Computer Systems
School of Electrical & Computer Engineering
National Technical University of Athens
Athens, Zografou, 15780, Greece
estai@netmode.ntua.gr

John S. Baras
Institute for Systems Research
Department of Electrical & Computer Engineering
University of Maryland
College Park, MD 20742, USA
baras@umd.edu

Symeon Papavassiliou
Institute of Communications and Computer Systems
School of Electrical & Computer Engineering
National Technical University of Athens
Athens, Zografou, 15780, Greece
papavass@mail.ntua.gr

ABSTRACT

Future communications consist of an increasing number of wireless parts, while simultaneously need to support the widespread multimedia applications imposed by social networks. These human-machine systems, driven by both real time social interactions and the challenges of the wireless networks' design, call for efficient and easy to implement, distributed cross-layer algorithms for their operation. Performance metrics such as throughput, delay, trust, energy consumption, need to be improved and optimized aiming at high quality communications. We investigate the coveted throughput-delay trade-off in static wireless multihop networks based on a "computer-aided" design of the backpressure scheduling/routing algorithm for networks embedded in hyperbolic space. Both routing and scheduling exploit the hyperbolic distances to orient the packets to the destination and prioritize the transmissions correspondingly. The proposed design provides us with the freedom of controlling its theoretical throughput optimality and of counterbalancing its practical performance through simulations, leading to significant improvements of the throughput-delay trade-off.

Categories and Subject Descriptors

C.2 [**Computer Systems Organization**]: Computer - Communication Networks; C.2.1 [**Computer - Communication Networks**]: Network Architecture and Design—*Wireless Communication, Network Communications, Distributed Networks*

General Terms

Performance, Algorithms, Design

Keywords

Wireless Networks, Greedy Embedding, Hyperbolic Geometry, Delay, Throughput

1. INTRODUCTION

The new era of networks integrates increasingly the wireless communications either as autonomous wireless network topologies or as parts of hybrid network infrastructures. In addition, we are witnessing the spread of multimedia applications imposing real-time traffic over wireless networks and calling for high throughput with low packet delays. Even more, the emerging development of social network applications over wireless ad hoc topologies strengthens the calling for efficient throughput-delay trade-offs.

The backpressure algorithm, introduced in its original form in [14], satisfies the requirement of throughput optimality. It performs routing and scheduling based on congestion gradients, by allowing transmission to the links that maximize the sum of differential queue backlogs in the network. However, by deploying routing without any information about the position or distance to the destination, the backpressure explores all possible source-destination paths leading to high delays. Several approaches have been developed towards the direction of reducing the delay imposed by the pure backpressure algorithm. To begin with, the authors in [15] combine backpressure and shortest path routing by imposing hop-count constraints on each flow, assuming that each node knows a-priori its hop distance from all others. Following another approach, the authors in [3] use shadow queues for the backpressure scheduling/routing, improving in this way the delay of the backpressure algorithm while simultaneously reducing the number of real queues need to be stored at each node. In the same spirit of "virtual" queues, the authors in [8] examine the problem of delay reduction in stochastic network optimization problems. After proving the relation of the backlog vector with the dual optimal solution (Lagrange multiplier) of the optimization problem, they use this finding to develop the FQLA algorithm that uses virtual place holder bits in order to reduce delays. Last but not least, in [9], a backpressure algorithm based on packet delays instead of queue backlogs is developed and shown to keep the average per flow delays well concentrated around their average values and to respond to the problem of finite flows.

In this paper, we propose a "computer-aided" backpressure algorithm which admits a static network embedded in hyperbolic space according to a greedy embedding. A greedy embedding in hyperbolic space is a correspondence between nodes and hyperbolic coordinates such that the greedy routing algorithm, employed in hyperbolic space, does not have local minima, i.e. every node can find at least one neighbor closer than itself to all possible destinations [4, 12]. We consider a greedy embedding of the network following [4] and we impose routing constraints on the backpressure algorithm, by determining as possible next-hop neighbors for a specific destination only "greedy" neighbors, i.e. those that strictly reduce the hyperbolic distance to the destination. Furthermore, we scale the differential backlogs on the links with a properly designed function of hyperbolic distances, in order to incorporate the hyperbolic distances in the scheduling procedure. As it is shown through simulations, this approach reduces the average per flow delay and increases the average per flow throughput for each examined network topology and social flow. The term "computer-aided" is used due to the derivation of the form of the scaling function through simulations. The basic intuition behind our idea lies in the occasion of specific network topologies such as the wireless multihop network with clique interference constraints, where it is delay-optimal to schedule the packets being closest to the destination [7]. We therefore believe that we should take into account the topological properties of the network together with the backlog information, in order to improve the delay performance.

The rest of the paper is organized as follows. In section 2, we describe the system model and basic knowledge regarding hyperbolic space and greedy embeddings. In section 3, we describe the proposed algorithm, prove its throughput optimality and discuss its complexity and distributivity. In section 4, we provide a numerical study for choosing the scaling function and illustrating the improvement in the throughput-delay trade-off, while possible directions for extensions to dynamic networks are briefly presented. Finally, section 5, concludes the paper.

2. SYSTEM MODEL AND HYPERBOLIC EMBEDDING

The system consists of a wireless multihop network with N nodes that function in a time-slotted fashion. Let us denote as $\mathbf{N}$ the set of the network nodes. At each time slot t, the scheduling/routing algorithm decides which set of non-interfering links is going to transmit and which flows will be served by each link, while new packets may arrive at the nodes. The number of packets that arrive at node i for destination d is a random variable $A_i^d(t)$, i.i.d. in all t, with expected value λ_i^d. We suppose that each node i stores a queue $q_i^d(t)$ for each destination d. We denote with $\mu_{ij}(t)$ the packets served by the link (i,j) and with the $N \times N$ matrix, $[\mu_{ij}(t)]$, the traffic over all the network, at time t. Also, we denote with $\mu_{ij}^d(t)$ the communication traffic on the link (i,j) for destination d at time t. The arrival and service rates are considered bounded and $\lambda_i^d \leq A_{\max}$. We use the term I_S to refer to the set of service rate matrices of all possible independent sets of the graph, i.e. maximal sets of links that do not interfere with each other. Also, we use the notation $\mathcal{N}(i)$ for the one-hop neighborhood of node i.

The whole infinite hyperbolic plane can be represented inside the finite unit disc $\mathcal{D} = \{z \in \mathcal{C} | |z| < 1\}$ of the Euclidean space; the Poincaré Disc model. The greedy embedding used in this work is based on the Poincaré Disc model. The hyperbolic distance function $d_H(z_i, z_j)$, for two points z_i, z_j, in the Poincaré Disc model is given by [4, 1]:

$$\cosh d_H(z_i, z_j) = \frac{2|z_i - z_j|^2}{(1-|z_i|^2)(1-|z_j|^2)} + 1 \tag{1}$$

The Euclidean circle $\partial\mathcal{D} = \{z \in \mathcal{C} | |z| = 1\}$ is the boundary at infinity for the Poincaré Disc model. In addition, in this model, the shortest hyperbolic path between two nodes is either a part of a diameter of $\mathcal{D}$, or a part of a Euclidean circle in $\mathcal{D}$ perpendicular to $\partial\mathcal{D}$.

The greedy embedding is constructed by choosing a spanning tree of the graph of the initial network and then embedding the spanning tree into the hyperbolic space according to the distributed algorithm of [4]. If a spanning tree of the graph admits a greedy embedding in the hyperbolic space then the whole graph admits also a greedy embedding [12]. Let us denote as "greedy" paths, the paths consisting of nodes with strictly decreasing hyperbolic distances to their destinations. From definition, the greedy embedding ensures the existence of at least one greedy path between each source-destination pair in the case of static networks. Every pair of nodes i, j is connected through a unique path, let us denote it as $i,\ i_1,\ i_2, \ldots\ i_k,\ j$, lying on the spanning tree which is embedded in the hyperbolic space. Due to the particular embedding, i_1 is at least one greedy neighbor of i for j and i_k is a greedy neighbor of j for i. Let us denote as $dist_H(i,d)$ the hyperbolic distance between nodes i, d.

We adapt the capacity region of [6], so as to include the routing constraints of the proposed class of algorithms. Therefore, the capacity region should allow routing only through greedy paths. The capacity region Λ_G is the set of all input rate matrices (λ_i^d) with $\lambda_i^d \neq 0$ if $i \neq d$ and (i,d) is a source destination pair, such that there exists a rate matrix $[\mu_{ij}]$ satisfying the following constraints:

- Efficiency constraints: $\mu_{ij}^d \geq 0$, $\mu_{ii}^d = 0$, $\mu_{dj}^d = 0$, $\sum_d \mu_{ij}^d \leq \mu_{ij}$, $\forall\ i, d, j$.
- Flow constraints: $\lambda_i^d + \sum_l \mu_{li}^d \leq \sum_l \mu_{il}^d$, $\forall\ i, d:\ i \neq d$.
- Routing constraints: $\mu_{ij}^d = 0$ if i has at least one greedy neighbor for d and j is not one of the $i's$ greedy neighbors.

Finally, we define the notion of strong stability of the queues, which will be used in the proof that follows. According to the Definition 3.1 of [6], a queue, $q_i^d(t)$ is strongly stable if $\limsup_{t\to\infty} \frac{1}{t}\sum_{\tau=0}^{t-1} E(q_i^d(\tau)) < \infty$. If all the queues of the network are strongly stable, then the whole network is strongly stable.

3. GREEDY BACKPRESSURE WITH QUEUE DIFFERENCE SCALING

In this section, we describe the proposed algorithm, which modifies the classic backpressure algorithm by performing routing only through greedy paths, and scheduling considering, in addition to the queue backlogs, the hyperbolic distances to the final destination of the transmitting and receiving nodes. The scheduling process weights the queue difference for destination d of each link (i,j) by a function $f(dist_H(i,d), dist_H(j,d)) : \mathbf{N}^3 \to \Re,\ i \neq j, d$, thus assigning priorities to particular links for transmission, choosing as crucial links for delay reduction those being highly prioritized. Algorithm 1 describes in pseudo-code the proposed scheduling/routing algorithm.

Algorithm 1: Greedy Backpressure with Queue Difference Scaling (GBP QDS)

for *each directed link* (i,j) **do**
  for *each destination* d **do**
    %Greedy backpressure%
    if $dist_H(i,d) > dist_H(j,d)$ **then**
      %Queue Difference Scaling
      $P_{ij}^d(t) = f(dist_H(i,d), dist_H(j,d))(q_i^d(t) - q_j^d(t))$;
    else
      $P_{ij}^d(t) = -\infty$;
  %Define the weight $P_{ij}(t)$ *as follows* :
  $P_{ij}(t) = \max(\max_d P_{ij}^d(t), 0)$;
  $d^*(i,j) = arg\max_d P_{ij}^d(t)$;
Choose the rate matrix through the maximization :
$[\mu_{ij}(t)] = arg_{\mu' \in \mathbf{I_S}} \max \sum_{(i,j)} \mu'_{ij} P_{ij}(t)$
for *each directed link* (i,j) **do**
  if $\mu_{ij}(t) > 0$ **then**
    the link (ij) *serves* $d^*(i,j)$ *with* $\mu_{ij}^{d^*}(t) = \mu_{ij}(t)$;
  For $d \neq d^*$ *we set* $\mu_{ij}^d(t) = 0$

It is important to mention that the greedy embedding ensures the existence of a greedy path for a source destination pair. So the routing constraints of the Algorithm 1 are well defined and there are no holes that can stack the packets to a specific node. Therefore, the correctness of the routing in terms of leading the packet to the destination is ensured. The function $f(dist_H(i,d), dist_H(j,d)) = f(i,j,d)$ is not arbitrarily defined and should satisfy some properties. More precisely, it should be a positive function. Due to the finiteness of the network we can define minimum and maximum values for $f(i,j,d)$, $f_{\min} \leq f(i,j,d) \leq f_{\max}$, where the condition $f_{\min} > 0$ should apply. As we will see from simulations, different types of functions $f(i,j,d)$ are suitable for different network topologies or types of social flows. However, the idea of the "hyperbolic distance-based" scheduling remains common in all different cases of physical topologies and social flows. In the case of $f(i,j,d) = 1$, let us denote the GBP QDS algorithm simply as GBP.

The Queue Difference Scaling approach provides us with freedom in choosing how to prioritize transmissions based on the network topology. As will be seen from simulations, it appears to behave uniformly over different network topologies and social flows, in the sense that we can find a function $f(i,j,d)$ that improves the throughput-delay trade-off, in all the examined cases. As a trade-off, the stability of queues under this approach can be easily proved only for a subset of the capacity region Λ_G, which depends on the values $f_{\max}, f_{\min}$ and leads to throughput optimality (the whole Λ_G) when $f_{\max} \to f_{\min}$ (Theorem 1). Finally, in the Queue Difference Scaling Approach, only links with positive differential backlog $q_i^d(t) - q_j^d(t)$ can transmit as in the classic backpressure algorithm.

THEOREM 1. *Let us define* $\epsilon_0 = \frac{f_{\max} - f_{\min}}{f_{\min}} A_{\max}$. *If we assume that the average arrival rates* $\lambda_i^d + \epsilon, \epsilon > \epsilon_0$ *lie inside the capacity region* Λ_G, *then the queues of the network are strongly stable, under the Greedy Backpressure with Queue Difference Scaling algorithm (GBP QDS) for static networks.*

PROOF. We define two indicator functions:
$I_1 = \{dist_H(i,d) > dist_H(j,d) \bigwedge j \in \mathcal{N}(i)\}$,
$I_2 = \{dist_H(i,d) < dist_H(j,d) \bigwedge i \in \mathcal{N}(j)\}$,
We use $|I_1$ to denote that the I_1 condition is satisfied for nodes i, j. Similarly for I_2. The queue dynamics in the case of Algorithm 1 are

$$q_i^d(t+1) = \max\{q_i^d(t) - \sum_{j|I_1} \mu_{ij}^d(t), 0\} + \sum_{j|I_2} \mu_{ji}^d(t) + A_i^d(t). \quad (2)$$

We denote by $Q(t) = [q_i^d(t)]$, the column array of the queues of the network. From the Lemma in page 52 of [6], we have that

$$(q_i^d(t+1))^2 \leq (q_i^d(t))^2 + \left(\sum_{j|I_1} \mu_{ij}^f(t)\right)^2 + \left(\sum_{j|I_2} \mu_{ji}^d(t) + A_i^d(t)\right)^2 - 2q_i^d(t)\left(\sum_{j|I_1} \mu_{ij}^d(t) - \sum_{j|I_2} \mu_{ji}^f(t) - A_i^d(t)\right). \quad (3)$$

As aforementioned, $f_{\min} \leq f(i,j,d) \leq f_{\max}$. The backpressure scheduling under the Queue Difference Scaling approach, is
$[\mu_{ij}(t)] = arg_{\mu' \in \mathbf{I_S}} \max \sum_{(i,j,d)} f(i,j,d) \mu_{ij}^{d\,\prime} \left(q_i^d(t) - q_j^d(t)\right)$.
Due to this maximization each summand of the form $f(i,j,d)\mu_{ij}^d(t)\left(q_i^d(t) - q_j^d(t)\right)$, $f(i,j,d) > 0$, with optimal $\mu_{ij}^d(t)$, is non negative or in other words, $\mu_{ij}^d(t) = 0$ whenever $q_i^d(t) - q_j^d(t) < 0$. Therefore, we can write that

$$\begin{aligned}
&\sum_{(i,j,d)} f_{\min} \mu_{ij}^d(t)\left(q_i^d(t) - q_j^d(t)\right), \\
&\leq \sum_{(i,j,d)} f(i,j,d) \mu_{ij}^d(t)\left(q_i^d(t) - q_j^d(t)\right), \\
&\leq \sum_{(i,j,d)} f_{\max} \mu_{ij}^d(t)\left(q_i^d(t) - q_j^d(t)\right),
\end{aligned} \quad (4)$$

where the service rates $\mu_{ij}^d(t)$ are chosen by the maximum weight matching (scheduling as in Algorithm 1). Also,

$$\begin{aligned}
&\max_{\mu' \in \mathbf{I_S}} \sum_{(i,j,d)} f_{\min} \mu_{ij}^{d\,\prime}\left(q_i^d(t) - q_j^d(t)\right), \\
&\leq \max_{\mu' \in \mathbf{I_S}} \sum_{(i,j,d)} f(i,j,d) \mu_{ij}^{d\,\prime}\left(q_i^d(t) - q_j^d(t)\right), \\
&\leq \max_{\mu' \in \mathbf{I_S}} \sum_{(i,j,d)} f_{\max} \mu_{ij}^{d\,\prime}\left(q_i^d(t) - q_j^d(t)\right).
\end{aligned} \quad (5)$$

We define the Lyapunov function $L(Q(t)) = \sum_{i,d} f_{\max} q_i^d(t)^2$

and we compute the Lyapunov Drift

$$E(L(Q(t+1)) - L(Q(t))|Q(t)) = E\left[\sum_{i,d} f_{\max} q_i^d(t+1)^2 \,\middle|\, Q(t)\right]$$

$$-E\left[\sum_{i,d} f_{\max} q_i^d(t)^2 \,\middle|\, Q(t)\right] \overset{Eq.(2,3)}{\leq} \sum_{i,d} f_{\max} q_i^d(t)^2 +$$

$$\sum_{i,d} f_{\max}\left\{E\left[\left(\sum_{j|I_1}\mu_{ij}^d(t)\right)^2 + \left(\sum_{j|I_2}\mu_{ji}^d(t) + A_i^d(t)\right)^2 \,\middle|\, Q(t)\right]\right\}$$

$$-\sum_{i,d} f_{\max}\left\{2q_i^d(t)E\left[\left(\sum_{j|I_1}\mu_{ij}^d(t) - \sum_{j|I_2}\mu_{ji}^d(t) - A_i^d(t)\right) \,\middle|\, Q(t)\right]\right\}$$

$$-\sum_{i,d} f_{\max} q_i^d(t)^2 \leq f_{\max}B + 2\sum_{i,d} f_{\max} q_i^d(t)\lambda_i^d$$

$$-2\sum_{i,d} f_{\max} q_i^d(t) E\left[\left(\sum_{j|I_1}\mu_{ij}^d(t) - \sum_{j|I_2}\mu_{ji}^d(t)\right) \,\middle|\, Q(t)\right], \quad (6)$$

where B is an upper bound of $\sum_{i,d} E\left[\left(\sum_{j|I_1}\mu_{ij}^d(t)\right)^2 + \left(\sum_{j|I_2}\mu_{ji}^d(t) + A_i^d(t)\right)^2 \,\middle|\, Q(t)\right]$. Therefore, we conclude to the following inequality

$$E(L(Q(t+1)) - L(Q(t))|Q(t)) \leq f_{\max}B + 2\sum_{i,d} f_{\max} q_i^d(t)\lambda_i^d$$

$$-2\sum_{i,d} f_{\max} q_i^d(t) E\left[\left(\sum_{j|I_1}\mu_{ij}^d(t) - \sum_{j|I_2}\mu_{ji}^d(t)\right) \,\middle|\, Q(t)\right],$$

$$= f_{\max}B + 2\sum_{i,d} f_{\max} q_i^d(t)\lambda_i^d$$

$$-2\sum_{d}\sum_{(i,j)|I_1} f_{\max}(q_i^d(t) - q_j^d(t))E\left[\mu_{ij}^d(t)\middle|Q(t)\right],$$

$$\overset{Eq.(4)}{\leq} f_{\max}B + 2\sum_{i,d} f_{\max} q_i^d(t)\lambda_i^d$$

$$-2\sum_{d}\sum_{(i,j)|I_1} f(i,j,d)(q_i^d(t) - q_j^d(t))E\left[\mu_{ij}^d(t)\middle|Q(t)\right]. \quad (7)$$

If $\lambda_i^d + \epsilon,\ \epsilon > \epsilon_0$ lie inside the capacity region, then from Corollary 3.9 in [6], there exist rates $\hat{\mu}_{ij}^d(t)$ determined according to the network topology and which, independently of the queue backlog, satisfy
$\lambda_i^d + \epsilon = E\left[\sum_{j|I_1}\hat{\mu}_{ij}^d(t) - \sum_{j|I_2}\hat{\mu}_{ji}^d(t)\right]\ \forall\, i,d.$
As a result, we have

$$\sum_{i,d} f_{\min} q_i^d(t)\lambda_i^d + \sum_{i,d} f_{\min} q_i^d(t)\epsilon$$

$$= \sum_{i,d} f_{\min} q_i^d(t) E\left[\sum_{j|I_1}\hat{\mu}_{ij}^d(t) - \sum_{j|I_2}\hat{\mu}_{ji}^d(t)\right],$$

$$= \sum_{d}\sum_{(i,j)|I_1} f_{\min} E\left[\hat{\mu}_{ij}^d(t)\right](q_i^d(t) - q_j^d(t)),$$

$$\leq \max_{\mu' \in \mathbf{I_S}} \sum_{d}\sum_{(i,j)|I_1} f_{\min}\mu_{ij}^{d\,\prime}(q_i^d(t) - q_j^d(t)),$$

$$\overset{(5)}{\leq} \max_{\mu' \in \mathbf{I_S}} \sum_{d}\sum_{(i,j)|I_1} f(i,j,d)\mu_{ij}^{d\,\prime}(q_i^d(t) - q_j^d(t)), \quad (8)$$

where $\hat{\mu}_{ij}^d(t)$ are the service rates chosen by the random scheduling algorithm. Therefore the Lyapunov drift (Eq. (7)) becomes

$$E[L(Q(t+1)) - L(Q(t))|Q(t)] \leq f_{\max}B$$

$$+2\sum_{d}\sum_{i} f_{\max}\lambda_i^d q_i^d(t) - 2\sum_{d}\sum_{i} f_{\min}(\lambda_i^d + \epsilon)q_i^d(t),$$

$$E[L(Q(t+1)) - L(Q(t))|Q(t)] \leq f_{\max}B$$

$$+2\sum_{d}\sum_{i}(f_{\max} - f_{\min})A_{\max}q_i^d(t) - 2\sum_{d}\sum_{i} f_{\min}q_i^d(t)\epsilon. \quad (9)$$

Therefore if the assumption $\epsilon > \epsilon_0 = \frac{f_{\max} - f_{\min}}{f_{\min}}A_{\max}$ is satisfied then from the Lemma 4.1 of [6], the network is strongly stable, as when $\sum_d \sum_i q_i^d(t) > \frac{f_{\max}B}{2(-(f_{\max} - f_{\min})A_{\max} + f_{\min}\epsilon)}$ the Lyapunov drift becomes negative. □

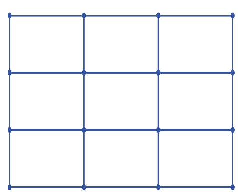
At this point, we should note that for the GBP algorithm, $f_{\min} = f_{\max} = 1$, thus Theorem 1 holds for $\epsilon > 0$, meaning that GBP is throughput optimal, ensuring stability for the entire capacity region Λ_G. Also, Theorem 1 is proved for a general positive function with finite value field, although in this paper we assume $f(i,j,d) = f(dist_H(i,d), dist_H(j,d))$.

3.1 Discussion on the Complexity and Distributivity of Algorithm 1

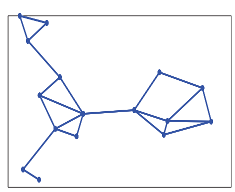
In this section, we briefly discuss the issues of the complexity and the distributivity of the Algorithm 1, mostly compared to those of the pure backpressure algorithm. The lack of centralized infrastructure in wireless multihop networks leads to the search of algorithms characterized by low-complexity to be implemented distributively. The proposed algorithm, has at most the same complexity as the original backpressure, due to the fact that each node computes fewer queue differences by checking at each neighboring link only the destinations for which this link is on a greedy path. Regarding the creation of the greedy embedding and the computation of the hyperbolic distances, they can be performed only once and stored at each node, and therefore they do not add any extra complexity at each step of the execution of the algorithm. Furthermore, the greedy embedding of [4] is based on a distributed algorithm where the hyperbolic coordinates are computed sequentially from the root to the leaves of the spanning tree. Additionally, with respect to the distributivity issue, the proposed algorithm if combined with a CSMA based backpressure scheme [10] achieves a completely distributed implementation, a fact that is also true for the classic backpressure algorithm. The method followed in [10] can be modified, so that the transmission aggressiveness of the CSMA algorithm is characterized by the product of the $f(i,j,d)$ and the queue difference instead of only the queue difference. Finally, several approximations of the maximum weight matching [5] which constitutes the caveat in the distributivity of the backpressure-style algorithms, can also be applied to Algorithm 1.

4. NUMERICAL EVALUATION

In this section, we study the delay-throughput trade-off performance in different topology and traffic scenarios. We examine two different physical layer topologies for the wireless multihop network, each one consisting of $N = 16$ nodes (Fig. 1). The first one is a 4x4 grid topology (Fig. 1(a)), while the second constitutes a Random Geometric Graph (RGG), where the nodes are randomly and uniformly distributed in a square region of side $L = 4m$ (Fig. 1(b)). Each link can transmit one packet during a time slot under the one-hop interference model. Throughput is expressed as the percentage of packets that reach their destination divided by those sent from the source for each flow and both throughput and delay are expressed as averages for all flows (source-destination pairs). Considering the social network of flows which is developed over our wireless network, three cases are simulated. The first one is denoted as the "All Pair Communication Model" where each node at each time slot chooses a random node and generates traffic for it with probability ranging from $\lambda = 0.01$ to $\lambda = 0.3$ with step increase 0.01. For each λ we run the algorithms for 5000 slots. The second examined social graph, is inspired by Kleinberg's model [11]. According to this graph each node communicates with all its one-hop physical neighbors and with one more node in further hop-distance, i.e. long-range social contact. At each time slot, each node chooses randomly one of its possible destinations and generates traffic for it with probability ranging from $\lambda = 0.01$ to $\lambda = 0.3$ with step increase 0.01. We call this model the "Kleinberg Model". Finally, the third type of social graph is denoted as the "Long Range Model", according to which each node communicates with 3 other long-distant (not one-hop neighbors) in physical layer nodes, chosen randomly. The traffic generation for the latter model is similar with the Kleinberg Model. According to previous research, the matching models for the social traffic over wireless networks are the Kleinberg Model or the Long Range Model [2]. In all simulations, we employ the Backpressure (BP), the GBP and the GBP QDS algorithms. Let us denote as $d_{\max}$ the largest hyperbolic distance among all node pairs in the network. Indeed, we need to note that each hyperbolic distance has a unique value.



(a) Grid Topology



(b) Random Geometric Graph Topology

Figure 1: Physical Layer Topologies.

The range of λ is chosen suitably from simulations in order to avoid excessive increase in the sum of queue lengths in the network under all physical and social topologies and algorithms, which would indicate the transgression of the capacity region. An example of the queue scaling for the All Pair Communication Model in the RGG topology is depicted in Fig. 2. We observe that after a certain value of λ, which concides with the value of λ at which the degradation in the throughput and delay begins (as shown in Fig. 6), the sum of queue backlogs in the network starts to increase, indicating that the network capabilities have been exceeded. The functions f_1, f_2, f_3 in Fig. 2 will be explained in the sequel of this section. Similar results are obtained for the other combinations of physical and social graphs and determine the suitable range of values for λ. For the Long Range Model and the grid physical topology, the sum of queue lengths increases much faster with respect to the increase of λ and therefore, we use a more limited range for λ (Fig. 5).

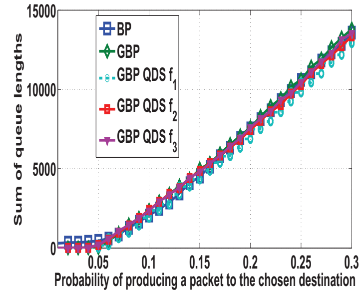


Figure 2: Queue Scaling for the RGG Topology - All Pair Communication Scenario.

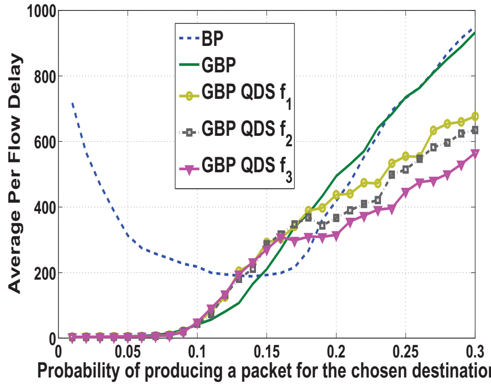
The combination of the grid topology and the All Pair Communication Model is examined first. In Fig. 3, the delay and throughput performance of the algorithms BP, GBP and GBP QDS for different chosen functions is illustrated. The employed functions are the following, $f_1 = |dist_H(i,d) - dist_H(j,d)|$, which gives higher weight to links that transfer the packets for longer hyperbolic distance, $f_2 = (2 \cdot d_{\max} - dist_H(i,d) - dist_H(j,d)) \cdot |dist_H(i,d) - dist_H(j,d)|$, which matches higher weights with links both close to the destination and with a high improvement of hyperbolic distance and $f_3 = \frac{|dist_H(i,d) - dist_H(j,d)|}{dist_H(i,d) + dist_H(j,d)}$, that has approximately the same meaning with f_2. It is important to note that the necessity of Greedy Routing constraints is reflected on the choices for the $f(i,j,d)$ function, as they ensure the correctness of the meaning of the $f(i,j,d)$, i.e. high difference $|dist_H(i,d) - dist_H(j,d)|$ implies moving closer in hyperbolic distance to destination. More precisely, due to the greedy routing constraints, we can ignore the absolute value and simply multiply by $(dist_H(i,d) - dist_H(j,d))$. We can observe that GBP QDS for f_1, f_2, f_3 performs better than GBP and BP, leading to higher throughput and lower delay, for all values of λ except from a small region $\lambda \simeq 0.14 - 0.18$. We conclude that GBP QDS with f_3 has a satisfactory behavior for this combination of physical topology and social communication graph.

Similarly with Fig. 3, in Fig. 4 the delay and throughput performance for Kleinberg's social graph over the grid topology, is shown. In the GBP QDS scheduling/routing the f_1 and f_3 scaling functions are examined. Similar conclusions as in the All Pair Communication Model apply, with a very good delay-throughput trade-off driven by the f_1 function for all values of λ.

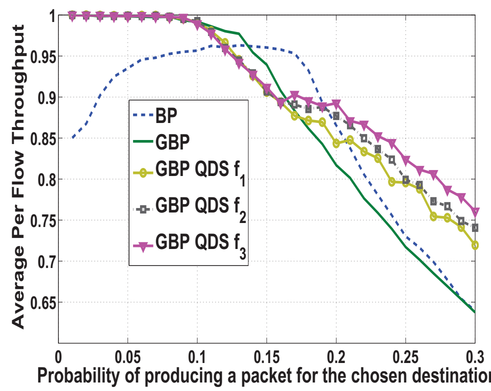
With respect to the combination of the grid topology and the Long Range Model, its performance is depicted in Fig. 5 where GBP QDS is applied with f_2, f_3 and $(f_3)^2$. Obviously, GBP QDS with $(f_3)^2$ leads to a significant improvement of the throughput-delay trade-off in the entire range of λ.

In the sequel, we examine the Random Geometric Graph physical topology, in order to pinpoint any differences in performance due to changes in the physical layer graph.

In Fig. 6, 7, 8, we depict the performance of the RGG physical topology and the three examined models of social flows. We employ the same functions as in the grid topology, with the exception of the Kleinberg Model, where the function $(f_3)^2$ achieves a better performance in terms of the

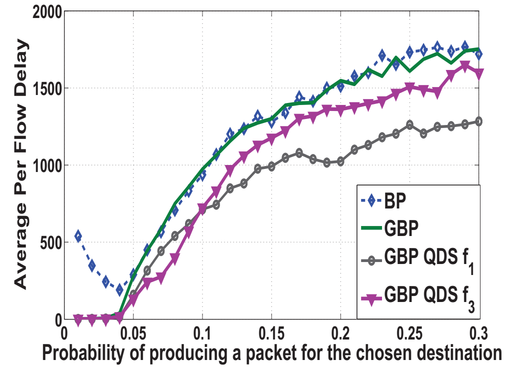


(a) Average Per Flow Delay

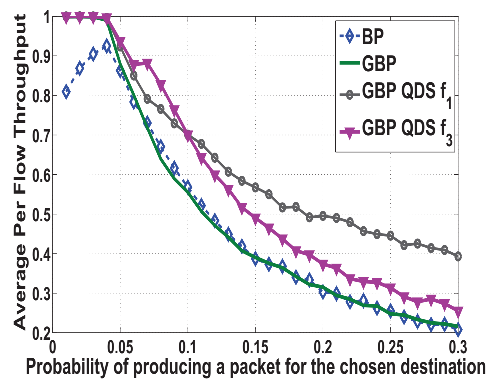


(b) Average Per Flow Throughput

Figure 3: GRID Topology - All Pair Communication Scenario.



(a) Average Per Flow Delay



(b) Average Per Flow Throughput

Figure 4: GRID Topology - Kleinberg's Communication Scenario.

throughput-delay trade-off. It can be observed that similarly with the grid topology, for each combination of the RGG topology and the three social models, we can find a function $f(i,j,d)$ that improves the throughput-delay trade-off of the pure backpressure algorithm.

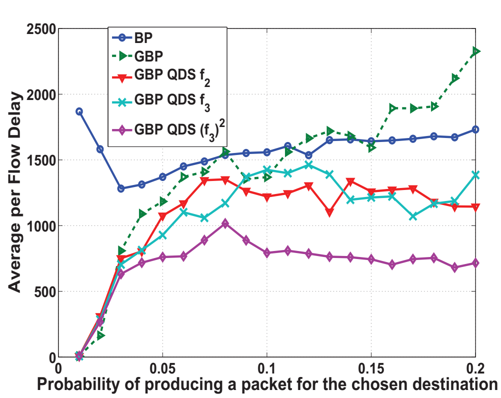
The GBP algorithm performs well in light traffic conditions due to routing the packets towards the direction of the destination and avoiding the long paths and cycles which characterize the classic backpressure algorithm. In higher traffic, it either presents the same or even sometimes worse performance than backpressure (depending on the social flows), as although orienting the packets towards the destination it restricts the number of paths creating more congested routes. However, the improvement in the throughput-delay trade-off is significant if topological properties in the hyperbolic space are taken into consideration in the scheduling part of the algorithm. Indeed, by combining greedy routing constraints with hyperbolic topology aware scheduling, we can improve the throughput-delay trade-off in a greater range of traffic values. In addition, from the simulations we can conclude that a good delay-throughput performance can be achieved by a function of the form

$$f(i,j,d) = \frac{(|dist_H(i,d) - dist_H(j,d)|)^{k_1}}{(dist_H(i,d) + dist_H(j,d))^{k_2}},\ k_1, k_2 = 0, 1, 2, \ldots \tag{10}$$

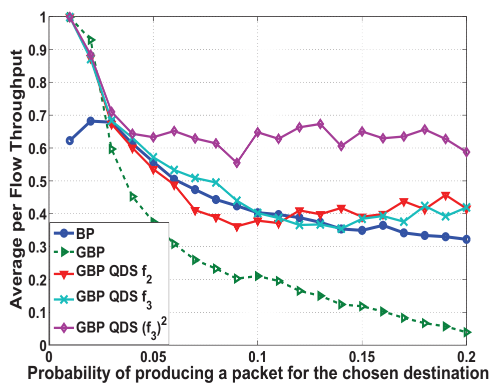
which can be used to initialize our searching for a suitable function in a different network topology and/or social flow. Specifically, we showed that this form of functions with $k_1 = k_2$ achieves a very good performance in our simulation scenarios. This class of functions, scales with $f_{\max} = 1$ the links adjacent to the destination and the rest of the links with diminishing values as we move further from the destination; depending also on the improvement that the link induces to the hyperbolic distance from the destination. This observation verifies our intuition to schedule with higher priority links closer to the destination.

4.1 Discussion on Extensions to Dynamic Networks

The Queue Difference Scaling approach of Algorithm 1 can be extended to dynamic networks, where the process of node churn is much slower than the routing/ scheduling process, by following the ideas proposed in our previous work [13]. In dynamic networks, new nodes join the network while existing nodes can resign from their network functionality. The greedy embedding algorithm of [4] allows for an easy and distributed integration in the greedy embedding of the newcomers. However, the deactivation of a node (supposing that the network still remains connected) can locally destroy the greedy embedding causing malfunctions to Algorithm 1. In this case, as proposed in [13] the GBP is replaced locally

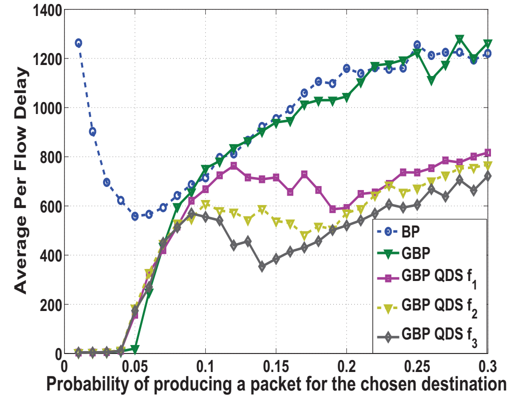


(a) Average Per Flow Delay

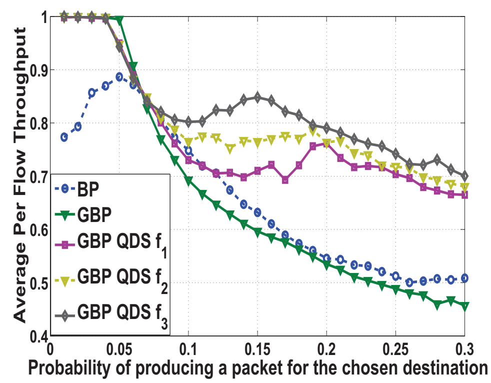


(b) Average Per Flow Throughput

Figure 5: GRID Topology - Long Range Communication Scenario.



(a) Average Per Flow Delay



(b) Average Per Flow Throughput

Figure 6: RGG Topology - All Pair Communication Scenario.

by the classic backpressure algorithm. However in the GBP QDS the function $f(i, j, d)$ needs to be redefined when the greedy property is locally lost, because, as aforementioned, its form may not express a valuable meaning for scheduling, i.e. f_3 is proportional to $|dist_H(i, d) - dist_H(j, d)|$ which is meaningful only if j is a greedy neighbor of i for destination d. Therefore, the function $f(i, j, d)$ should be replaced by a time dependent function $f(i, j, d, t)$, possible forms of which, at the times t when the greedy property between i, d is lost, will be examined in our future work.

5. CONCLUSIONS

In this work, we focused on the throughput-delay trade-off in static wireless networks, and for the scheduling/routing algorithms analyzed. We proposed a design of the backpressure algorithm in the hyperbolic space that scales the queue differences by a suitable function. Through rigorous analysis, we proved the stability of the GBP QDS algorithm for a part of the capacity region which can be controlled through the min and max properties of the scaling function. Through extended simulations, we illustrated the improvement in the throughput-delay trade-off, which is possible with a sophisticated choice of the scaling function and introduced a class of topology-dependent functions that achieve a satisfactory performance in the examined physical and social topologies. The proposed algorithm if combined with a CSMA based backpressure scheme [10] can lead to high quality performance via a completely distributed implementation.

6. ACKNOWLEDGMENTS

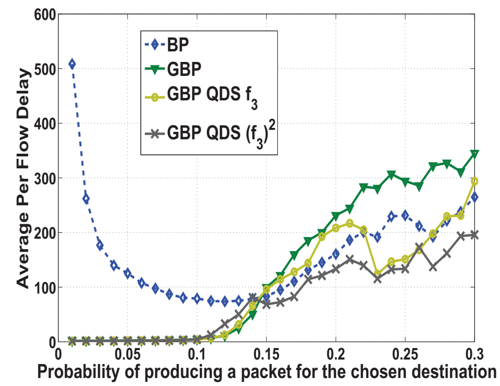
The research of Eleni Stai and John Baras was partially supported by the US Air Force Office of Scientific Research under MURI grant FA9550-09-1-0538, and by the National Science Foundation (NSF) under grant CNS-1018346. Eleni Stai was also partially supported by the Foundation for Education and European Culture (IPEP).

7. REFERENCES

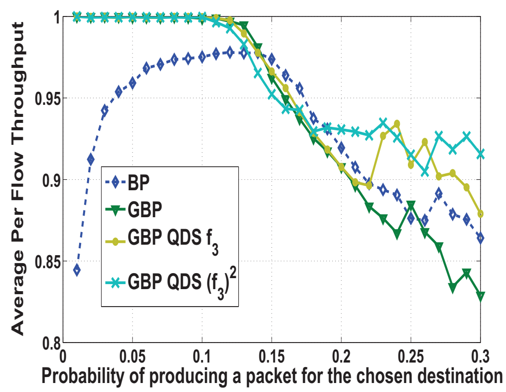
[1] J. W. Anderson. *Hyperbolic Geometry*. Springer 2nd Edition, 2007.

[2] B. Azimdoost, H. R. Sadjadpour, and J. J. Garcia-Luna-Aceves. Capacity of Composite Networks: Combining Social and Wireless Ad Hoc Networks. In *Proceedings of IEEE Wireless Communications and Networking Conference (WCNC)*, pages 464 – 468, March 2011.

[3] L. Bui, R. Srikant, and A. L. Stolyar. Novel Architectures and Algorithms for Delay Reduction in Back-pressure Scheduling and Routing. In *Proceedings of IEEE INFOCOM*, pages 2936 – 2940, April 2009.

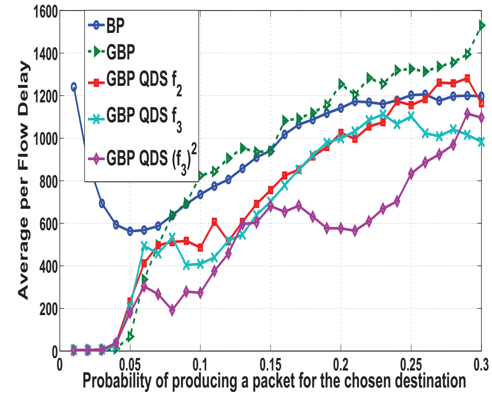


(a) Average Per Flow Delay

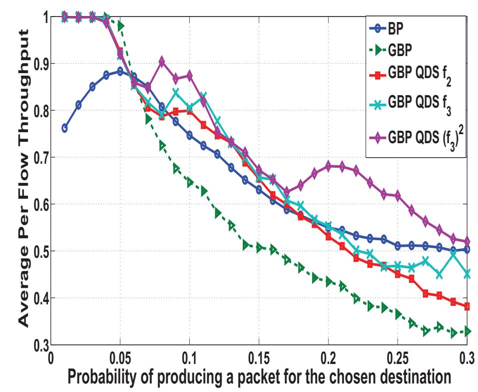


(b) Average Per Flow Throughput

Figure 7: RGG Topology - Kleinberg's Communication Scenario.



(a) Average Per Flow Delay



(b) Average Per Flow Throughput

Figure 8: RGG Topology - Long Range Communication Scenario.

[4] A. Cvetkovski and M. Crovella. Hyperbolic Embedding and Routing for Dynamic Graphs. In *Proceedings of IEEE INFOCOM*, pages 1647 – 1655, April 2009.

[5] A. Eryilmaz, A. Ozdaglar, D. Shah, and E. Modiano. Distributed Cross-layer Algorithms for the Optimal Control of Multi-hop Wireless Networks. *IEEE/ACM Transactions on Networking*, 18(2):638–651, April 2010.

[6] L. Georgiadis, M. J. Neely, and L. Tassiulas. *Resource Allocation and Cross-Layer Control in Wireless Networks*. Now Publications, 2006.

[7] G. R. Gupta and N. B. Shroff. Delay Analysis and Optimality of Scheduling Policies for Multihop Wireless Networks. *IEEE/ACM Transactions on Networking*, 19(1):129–141, February 2011.

[8] L. Huang and M. J. Neely. Delay Reduction via Lagrange Multipliers in Stochastic Network Optimization. *IEEE Transactions on Automatic Control*, 56(4):842–857, April 2012.

[9] B. Ji, C. Joo, and N. B. Shroff. Delay-based Back-pressure Scheduling in Multi-hop Wireless Networks. In *Proceedings of IEEE INFOCOM*, pages 2579 – 2587, April 2011.

[10] L. Jiang and J. Walrand. A Distributed CSMA Algorithm for Throughput and Utility Maximization in Wireless Networks. *IEEE/ACM Transactions on Networking*, 18(3):960–972, June 2009.

[11] J. Kleinberg. The Small-World Phenomenon: An Algorithmic Perspective. In *Proceedings of the 32nd ACM Symposium on Theory of Computing*, 2000.

[12] R. Kleinberg. Geographic Routing Using Hyperbolic Space. In *Proceedings. of IEEE INFOCOM*, pages 1902 – 1909, May 2007.

[13] E. Stai, J. S. Baras, and S. Papavassiliou. Throughput-Delay Trade-off in Wireless Multihop Networks via Greedy Hyperbolic Embedding. In *Proceedings of the 20th Int'l Symposium on Mathematical Theory of Networks and Systems (MTNS)*, July 2012.

[14] L. Tassiulas and A. Ephremides. Stability Properties of Constrained Queueing Systems and Scheduling Policies for Maximum Throughput in Multihop Radio Networks. *IEEE Transactions on Automatic Control*, 37(12):1936–1948, December 1992.

[15] L. Ying, S. Shakkottai, A. Reddy, and L. Shihuan. On Combining Shortest-path and Back-pressure Routing over Multi-hop Wireless Networks. *IEEE/ACM Transactions on Networking*, 19(3):841–854, June 2011.