An Optimization Framework for Opportunistic Multipath Routing in Wireless Mesh Networks

Božidar Radunović, Christos Gkantsidis, Peter Key, Pablo Rodriguez
Microsoft Research, Telefonica Research,
Cambridge, UK Spain

Abstract—We consider wireless mesh networks, and exploit the inherent broadcast nature of wireless by making use of multipath routing. We present an optimization framework that enables us to derive optimal flow control, routing, scheduling, and rate adaptation schemes, where we use network coding to ease the routing problem. We prove optimality and derive a primal-dual algorithm that lays the basis for a practical protocol. We use simulation to show on realistic topologies that we can achieve 20-100% throughput improvement compared to single path routing, and several times compared to a recent related opportunistic protocol (MORE).

Index Terms—wireless mesh networks, network coding, opportunistic routing, broadcast, multi-path routing, flow control, fairness, rate adaptation

I. INTRODUCTION

One of the main challenges in building wireless mesh networks ([2], [3], [4]) is to guarantee high performance. The difficulty is mainly caused by the unpredictable and highly-variable nature of the wireless channel. However, the use of wireless channels presents some unique opportunities that can be used to improve the performance. For example, the broadcast nature of the medium can be used to provide opportunistic transmissions as suggested in [5]. Also, in wireless mesh networks, there are typically multiple paths connecting each source destination pair; using some of these paths in parallel can improve performance [6], [7]. The optimal use of multiple paths and of opportunistic transmissions is the main focus of this work. We use network coding [8] to simplify the problem of scheduling packet transmissions across multiple paths, similarly to [6], [7], [9]. We propose a network optimization framework that optimizes the rate of packet transmissions between source and destination pairs.

In order to use the resources of the wireless mesh network efficiently, the system needs to take into account: (a) the existence of multiple paths, (b) the unreliable nature of the links, (c) the existence of multiple transmission powers and rates (which in turn affects the probability of correct packet reception), (d) the broadcast nature of the channel, (e) the competition among many flows, (f) fairness and efficiency. As we shall see, it is important to optimize all these parameters simultaneously to achieve optimal performance.

We use an optimization framework to design a distributed maximization algorithm. We account for transport layer controls and address questions of fairness by maximizing the aggregate utility of the end-to-end flows, where we associate a utility function $U(\cdot)$ with a flow. Because we use network coding, our optimization framework borrows heavily from [10], [9]. Our algorithm is a primal-dual algorithm [11]. The primal formulation expresses the optimization problem as a function of the rates of the various flows in the network; the dual formulation uses variables the queue lengths (per flow and per node). The main advantage of using the dual formulation of the optimization problem is that the dual variables (also referred as shadow prices) relate to queue lengths and can be directly used by back-pressure algorithms for flow control [12], [13]. As a simple example, a large number of queued packets for a particular flow at an internal node can be interpreted that the path going through that node is congested and should be avoided. The main advantage of using the primal-dual formulation is that it adapts the primal variables (i.e. flow rates) more slowly, hence, allows TCP-like window-based rate control modeling (as originally mentioned by Erylimaz et al. [12]). We propose a novel algorithm for cross-layer optimization and prove, using Lyapunov functions, that it converges to the optimal rate allocation.

The main contributions of our paper is a network wide primal-dual optimization algorithm that maximizes rate-based global network performance, and extends previous work by incorporating broadcast (opportunistic routing), multi-path routing, and fairness/rate control (Sections II and III). We introduce a notion of virtual packets, called credits, that enable us to decouple routing and flow control from actual packet transmissions. We prove the optimality of the algorithm. Using simulation on realistic topologies, we show we can achieve 20-100% throughput improvement with our distributed implementation compared to single path routing, and 20-300% compared to MORE [6] (Section IV).

II. MODEL

In this section we introduce the notation used in the paper. We extend the model of wireless erasure network developed in [14] to include multiple flows. Vectors are denoted in bold.

A. PHY and MAC Characteristics

We consider a network comprising of a set of nodes $\mathcal{N}$, $N = |\mathcal{N}|$. Whenever a node transmits a packet, several nodes may receive it. We model packet transmission from node $i$ to a
set of nodes $J \subseteq \mathcal{N}$ with a hyperarc $(i, J)$. We define an activation profile $S = \{S_i\}$ to be a set of hyperarcs active at the same time. There may be several constraints on feasible activation profiles. For example, a node may be limited to receive from but one node, or transmit to only one node at a time. The only condition we shall impose is that a node can be the source of only one hyperarc in one activation profile.

We denote by $S$ the set of feasible activation profiles and let $\text{SRC}(S) = \{i \in \mathcal{N} | \exists J \subseteq \mathcal{N}, (i, J) \in S\}$ be the set of transmitters in activation profile $S$.

Each transmission has two associated parameters, power $P \in \mathcal{P}$ and rate $R \in \mathcal{R}$, where $\mathcal{P}$ is the set of allowed transmission powers (e.g. $P \subseteq [0, P_M]$), where $P_M$ is given by regulations) and $\mathcal{R}$ is sets of available PHY transmission rates, defined by supported spreading, coding, and modulations.

Consider an activation profile $S$ in which node $i$ transmits to set of nodes $J$, and suppose node $i$ is transmitting with power $P_i$ and rate $R_i$. We can associate power vector $\mathbf{P} = (P_i)_{i \in \mathcal{N}}$ and rate vector $\mathbf{R} = (R_i)_{i \in \mathcal{N}}$ to these transmissions. Let $T_{ij} = 1$ if a packet is successfully transmitted from $i$ to $j$. We define $p_{ij}(P, R, S) = \text{Prob}(T_{ij}(P, R, S) = 1)$ to be the probability that node $j$ in $J$ will successfully receive the packet from $i$, given the above conditions. We also assume that $T_{ij}$ and $T_{ik}$ are independent for $i \neq l$ or $j \neq k$, which is justified by assumptions of (c.f. [15]). By convention, we assume $p_{ij}(P, R, S) = 0$ if $j \notin J$, for $(i, J) \in S$.

We can now calculate $C_{iK}(\mathbf{P}, \mathbf{R}_i, S)$ the average number of packets per unit time conveyed from node $i$ to any of the nodes in $K \subseteq \mathcal{N}$. We have

$$C_{iK}(\mathbf{P}, \mathbf{R}_i, S) = R_i \left(1 - \prod_{j \in K} (1 - p_{ij}(\mathbf{P}, \mathbf{R}_i, S))\right).$$

Note that $C_{iK}(\mathbf{P}, \mathbf{R}_i, S) = 0$ if $K \cup J = \emptyset$, for $(i, J) \in S$.

### B. Traffic, Routing and Flow Scheduling

There is a set of unicast end-to-end flows $\mathcal{C}$ in the network, and each flow $c \in \mathcal{C}$ has a source and a destination node $\text{Src}(c), \text{Dst}(c) \in \mathcal{N}$ respectively. Each flows may take an arbitrary route from a source to a destination, subject to constraints imposed by the set of activation profiles. We denote by $f_c$ the rate of flow $c \in \mathcal{C}$. One flow may take multiple routes to the destination. We assume routing is done at each hop. Each relay will decide how much traffic from a flow it will forward to other nodes. This decision is made through credit assignment, as described next in Section II-D.

Whenever a node is active, it needs to decide which flow it will transmit. This is defined through a flow-scheduling profile $\mathbf{A}$. If node $i$ transmits a packet from flow $c$ we set $A_{ic} = 1$, otherwise $A_{ic} = 0$. We say that a flow scheduling profile is valid if for each $i \in \mathcal{N}$ there exists only one $c \in \mathcal{C}$ such that $A_{ic} = 1$. Let $\mathcal{A}$ be the set of all valid flow scheduling profiles.

### C. Network Coding

We assume network coding per flow is used [14], [6]. The main benefit of network coding is that it facilitates scheduling.

Without network coding, when the same packet is received by several nodes, a mechanism is needed to prevent two or more nodes forward the same packets [5]. To eliminate this problem, each relay forwards a random linear combination of all previously received packets from the same flow.

#### D. Credits

Whenever a packet is transmitted, it may be received by several nodes, and it is important to decide which should forward packets, to avoid redundant transmissions (as explained in [16], [6]). We will use a concept of credits, which is similar to the control decision variable of Neely [16].

Credits are created for each packet at the source node. They are interpreted as the number of packets of a specific flow to be transferred by the node. Credits are conserved until they arrive at the flow’s destination. The main advantage of the credit scheme is that it simplifies scheduling. Credits are declarations of intent. The actual packet transmissions may occur at arbitrary time instants. Due to the use of network coding, we only need to ensure that the total number of packets transmitted between each two nodes corresponds to the number of credits. Thus, scheduling is done at a flow level and not at the packet level, incurring significantly smaller overhead.

As each credit delegates one packet to a node, we may express all the rates in the system in terms of credits. For example, $y_{ij}^c$ is the rate of credits of flow $c$ passed from node $i$ to node $j$. Theorem 1 shows that the rate of independent packets received at a destination of each flow will correspond to the number of credits delivered.

### E. Dynamics and Stability

We further assume the system is slotted in time. In each slot $t = 0, 1, \ldots$ a medium access protocol assigns an activation profile $S(t)$ and a flow-scheduling profile $\mathbf{A}(t)$, and to each transmitter $i \in \text{SRC}(S(t))$ we assign transmit power $P_i(t)$ and rate $R_i(t)$. We further denote by $y_{ij}^c(t)$ the number of credits for flow $c$ transmitted from node $i$ to node $j$ during slot $t$, and with $x_i^c(t)$ the number of packets of flow $c$ actually transmitted from $i$ to any of the nodes in $J$ during slot $t$. Let $f_c(t)$ be the number of fresh packets/credits generated at the source of flow $c$.

Note that, because each packet transmission is always associated with a credit transmission, we look at credit queues. Let $q_i^c(t)$ the amount of credits of flow $c$ queued at node $i$. The system is stable if every queue size is bounded. We will define stability more formally in Section III-D.

### III. OPTIMAL FLOW CONTROL FOR FAIRNESS

#### A. Feasible Rate Set

Assume an assignment of end-to-end rates $f_c$, for each flow $c$, and denote the rate vector by $\mathbf{f} = (f_c)_{c \in \mathcal{C}}$. The vector of rates is valid under the following three conditions. First, by conservation of the flow of credits at each node $i \notin \text{Dst}(c)$:

$$\sum_{j \neq i} y_{ij}^c + f_c 1_{i = \text{Src}(c)} \leq \sum_{j \neq i} y_{ij}^c$$

(2)
Also, $y_{ij} \geq 0$. Second, due to the constraints of the broadcast regions, for all $J \subseteq \mathcal{N}$ we have:

$$\sum_{j \in J} y_{ij} \leq x^e_{ij}$$ (3)

Assume now that variables $\alpha_{S,R,P,A}$ define a schedule and denote a fraction of time network uses scheduling profile $S$, routing profile $A$ and power and rate allocations $R, P$. By definition, $\alpha_{S,R,P,A} \geq 0$ and $\sum_{S,R,P,A} \alpha_{S,R,P,A} \leq 1$. The third condition comes from scheduling constraints:

$$x^e_{ij} \leq \sum_{S,A,R,P} \alpha_{S,R,P,A} A_{uc} C_{ij}(P, R_i, S)$$ (4)

Note that, although (4) implies $\{\sum_{c} x^e_{ij}\}_{i,j}$ belongs to Hull{$\{C_{ij}(P, R_i, S)\}_{i,j}$}, the converse is not true.

We will use the following characterization of feasible rates from [9]:

**Definition 1:** Vector $f$ is said to be feasible if each flow $c$ can transport information from Src$(c)$ to Dst$(c)$ at rate $f_c$.

**Theorem 1:** Let $F$ be the set of end-to-end rate vector $f = (f_c)_{c \in C}$ such that there exists vectors $y = (y_{ij})_{i,j \in \mathcal{N}, i \neq j \in C}$, $\alpha = (\alpha_{S,R,P,A})_{S,S,R,P,A \in A}$ that satisfy (2), (3), and (4) subject to $\alpha_{S,R,P,A} \geq 0$ and $\sum_{S,R,P,A} \alpha_{S,R,P,A} \leq 1$. The vector $f$ is feasible when coding generation size goes to infinity if and only if it belongs to $F$. Moreover, the set of feasible end-to-end rates $F$ is convex.

**Proof:** Follows directly from [9].

**B. Utility Maximization**

For each flow $c \in C$ we define a utility function $U_c(\cdot)$ to be a strictly concave, increasing function of end-to-end flow rate $f_c$. The utility of flow $c$ is then $U_c(f_c)$.

We can write the network-wide optimization problem as

$$\max_{f \in F} \sum_{c \in C} U_c(f_c)$$ (5)

Since set $F$ is convex and the objective is strictly concave, there exists a unique solution $f^*$ to the maximization problem. Corresponding $y^*$, $\alpha^*$ also exist but are not necessarily unique.

Let us denote with $\mu^c_i$, and $\xi^c_{ij}$ the Lagrangian multipliers associated with inequalities (2) and (3), respectively. To simplify the notation we will also define $\mu^c_{\text{Dtt}(c)} = 0$. We can write the KKT conditions at the optimal point

$$\mu^c_i \left( \sum_{j \neq i} y_{ij}^c - \sum_{j \neq i} y_{ji}^c - f^*_{c1} 1_{i=\text{Src}(c)} \right) = 0, \quad (6)$$

$$\xi^c_{ij} \begin{pmatrix} x^e_{ij} - \sum_{j \neq i} y_{ij}^c \\ f^*_{c1} - \mu^c_{\text{Src}(c)} \end{pmatrix} = 0, \quad (7)$$

$$f^*_{c} \begin{pmatrix} x^e_{ij} - \sum_{j \neq i} y_{ij}^c \\ \mu^c_{\text{Src}(c)} \end{pmatrix} = 0, \quad (8)$$

We see that $\mu^c_i$ can be positive only if more traffic of flow $c$ comes into node $i$ than leaves it. Hence intuitively we can relate $\mu^c_i$ to $q_i(t)$, the number of credits for flow $c$ queued at $i$. Similarly we can relate $\xi^c_{ij}$ to the number of packets queued for broadcasting at $i$. In Section III-C we will express this relationship more formally. We will also use (8) to develop a flow control algorithm.

As a consequence of KKT, using some elementary algebra one can derive

$$0 \geq \mu^c_i - \mu^c_j - \sum_{S \subseteq \mathcal{N}, j \in S} \xi^c_{ij}, \quad (9)$$

$$C^c_{ij} = \arg \max_{C_{ij}(P, R_i, S)} \sum_{i} \max_{c} \sum_{j \in S} \xi^c_{ij} C_{ij} \quad (10)$$

Notably we will use (10) in Section III-C to derive the optimal scheduling.

**C. Maximization Algorithm**

We next present an algorithm that converges to the optimal value of (5). In the following we assume that the feedback is ideal, hence that the acknowledgments and credits are transmitted instantaneously and without errors. We leave the analysis of signaling with losses and delays for future work.

**Node and Transport Credits:** Recall that $q_i^c(t)$ is the amount of credits of commodity $c$ queued at node $i$. We call these credits node credits. In addition, let $w_{ij}^c(t)$ be the number of credits of commodity $c$ queued at $i$ and corresponding to the packets that have to be delivered to any of the nodes in $J$ (as previously decided by the credit transmission scheme). We call these credits transport credits. When a credit for flow $c$ is passed from node $i$ to node $j$, we decrease $q_i^c$, we increase $q_j^c$, and we increase $w_{ij}^c$ for all $J \ni j$ (all of them by one unit). We decrease $w_{ij}^c$ when a packet from flow $c$ is actually transmitted from $i$ to any of the nodes in $J$.

**Routing protocol:** Node credits represent intentions of packet transmissions and a routing protocol describes when and how are node credits transferred. Let $y_{ij}^c(t)$ be the number of node credits for flow $c$ transferred from node $i$ to node $j$ at time $t$ and let us define $w_{ij}^c(t) = \sum_{X \subseteq \mathcal{N}, j \in X} q_i^c(x, t)$. A back-pressure between nodes $i$ and $j$ is defined as

$$z_{ij}^c(t) = q_i^c(t) - w_{ij}^c(t) - q_j^c(t),$$

the difference between the excess credits queued of flow $c$ at node $i$ not destined for node $j$ ($q_i^c - w_{ij}^c$) and the node credits at node $j$ ($q_j^c$). A credit is routed from $i$ to $j$ only if the back-pressure is positive:

$$y_{ij}^c(t) = M 1_{\{z_{ij}^c(t) > 0\}}, \quad (11)$$

where $1_{\{x > 0\}}$ is 1 if $x > 0$ or 0 otherwise. In Section III-D we will derive conditions on $M$ to guarantee convergence of the algorithm.

**Scheduling, rate and power control:** The optimal scheduling, rate and power control algorithm is the tuple $(S(t), P(t), R(t), A(t))$ that solves the following optimization
problem
\[ \bar{w}_i(t, \mathbf{P}, R_i, S) = \max_c \sum_j w_i^c j(t) C_{iJ}(\mathbf{P}, R_i, S) \] (12)
\[ (S(t), \mathbf{P}(t), \mathbf{R}(t)) = \arg \max_{S, \mathbf{P}, \mathbf{R}} \sum_{i \in \mathcal{N}} \bar{w}_i(t, \mathbf{P}, R_i, S), \] (13)
\[ C_{iJ}(t) = C_{iJ}(P(t), R_i(t), S(t)), \] (14)
\[ c^*_{iJ}(t) = \arg \max_c \sum_K w_i^K C_{iK}(t), \] (15)
\[ A_{ic}(t) = 1_{\{c = c^*_{iJ}(t)\}}, \] (16)
\[ x_{iJ}^*(t) = A_{ic} C_{iJ}(t) \] (17)

Equations (12)-(17) represent a joint scheduling, rate, and power control problem. We find the optimal scheduling, power and rate control problem \((S(t), P(t), R(t))\) by solving (13). Then, equation (15) is used to select which flow will be transmitted by each node in slot \(t\). Note that we cannot decouple the flow selection process \(A(t)\) and routing/scheduling/rate/power control as it was done in similar approaches that do not use opportunistic routing (e.g. \([17, 18, 12]\)). Also, unlike in \([17, 18, 12, 16]\), we do not explicitly use back-pressure information for scheduling in (12)-(17); instead we use transmission credits \(w_i^c j(t)\).

**Flow control:** The optimal flow rate at the source, \(f^c_i(t)\) can be calculated using a primal-dual approach, as in [12]
\[ f_c(t + 1) = \left[f_c(t) + \gamma \left(U_c^f(f_c(t)) - q_{\text{Src}(c)}(t)\right)\right]^+, \] (18)
where \([x]^+ = \max\{x, 0\}\). Each flow adapts its rate based on the previous rate and current number of credits queuing for transmission at the source node for that flow \(q_{\text{Src}(c)}\). The primal-dual approach well describes additive-increase multiplicative-decrease transport protocols, like TCP [11].

**D. Convergence Of The Algorithm**

Proof of the convergence of the algorithm can be found in our technical report [1].

**E. Comparison with MORE**

In this section we compare the performance of our algorithm with the MORE algorithm presented in [6], [19]. Our algorithm converges to the optimal solution of the optimization problem \((5)\) (as shown in [1]), hence MORE at best is as good as our algorithm. We first show under what conditions MORE is guaranteed to give the optimal solution. We then also illustrate by two examples that MORE can yield strictly suboptimal rate allocations.

**Theorem 2:** If there is only one flow in the system, if transmission rates and powers of all nodes are fixed and if only one node can transmit at a time (that is \(\text{SRC}(S) = 1\) for all \(S \in \mathcal{S}\)) MORE and our algorithm give the same performance.

**Proof:** Since only one node can transmit at a time, we have \(S = \mathcal{N}\). Furthermore, transmission powers and rates are fixed, hence (3) and (4) reads as \(\sum_{i \in J} y_{ij} \leq \alpha_i C_{iJ}\). We also omit \(c\) as there is only one flow in the system.

We start with the MORE optimization problem, as given in [19], and we introduce \(f = 1/\left(\sum_{i \in \mathcal{N}} z_i\right)\), \(y_{ij} = f x_{ij}\) and \(\alpha_i = f z_i\). The optimization from [19, Eq.(1)-(4)] is then equivalent to
\[ \min \frac{1}{f} \quad \text{s.t.} \quad \sum_{j \in \mathcal{N}} y_{ij} - \sum_{j \in \mathcal{N}} y_{ji} = f 1_{\{i = \text{Src}\}} \alpha_i C_{iJ} \geq \sum_{j \in \mathcal{N}} y_{ji}, \quad \sum_{i \in \mathcal{N}} \alpha_i = 1, \]
which is exactly the optimization problem \((5)\).

**IV. SIMULATION RESULTS**

We now present simulation results which quantify the performance advantages of the opportunistic routing, scheduling and flow control algorithms defined in the previous sections. We compared our algorithm with a conventional, single path routing algorithm, and with the MORE algorithm [6]. To make the comparison fair, we assumed that the single-path routing algorithm used the same kind of jointly-optimal routing and flow-control approach as our scheme, which boils down to [12].

In contrast, MORE does not integrate flow control or flow scheduling with the routing algorithm. When simulating the MORE algorithm, defined in [6], [19], we assumed that each source had a large backlog of packets to transmit, and that each relay performed FIFO scheduling among packets from different flows.

We used the roofnet network topology based on 802.11b cards, given in [5], for our simulations. Transmission probabilities between each pair of nodes for different transmission rates are given in [2]. We used \(U_c(\cdot) = \log(\cdot)\), hence the rate allocation that maximizes \((5)\) is the proportionally fair rate allocation [20].

We looked at two performance metrics. The first one is the improvement in total utility \(\sum_c U(f_c) - \sum_c U(f' c)\). Allocation \(f\) is better than \(f'\) if the sum difference is positive. The proportional fair rate maximizes the optimization problem \((5)\) hence has the highest utility. The second metric is the total rate improvement \(\sum_c f_c / \sum_c f'_c\). Allocation \(f\) is better than \(f'\) if the quotient is larger than 1. The proportionally fair allocation does not always have highest total rate.

We developed a discrete-event simulator that implements the three routing, flow and rate control algorithms. Since finding a solution to Equations (12)-(17) is an NP-hard problem, we use different heuristics to find a (sub)optimal solution. We ran simulations to obtain end-to-end rate allocations. We then ran the previous experiment with 100 random traffic matrices and compared the performances of the different algorithms with respect to the two performance metrics.

**Single vs. Multiple Paths:** In more than 80% of runs, our heuristic achieved higher total rate than the conventional, single-path algorithm. In more than half of the runs, the total rate has increased by 20%, and in some cases by over 100%. From these results we see that there is a significant advantage in using our multi-path routing algorithm over the single-path one. We see that the advantage is significant even in number of flows is large (see next paragraph for explanation). Our
This paper proposes an optimization framework for addressing questions of multi-path routing in wireless mesh networks. We have extended previous work by incorporating the broadcast nature of wireless and simultaneously addressing fairness issues. Implicit in our approach is the use of network coding, which enables us to define notions of credits that are used to track the number of transmitted packets, rather than specific packets themselves. Using our framework we show that our algorithm significantly outperforms single-path routing and MORE [6].

Our primal-dual rate adaptation can be used to model window-based flow control schemes, such as TCP. The performance of applications that run on top of our system and use TCP is an interesting open problem. Another interesting direction is to analyze the performance of our protocol with more realistic signaling schemes.

REFERENCES


