Distributed Algorithms for Multipath Routing in Full-Duplex Wireless Networks

Xi Fang
Arizona State University
xi.fang@asu.edu

Dejun Yang
Arizona State University
dejun.yang@asu.edu

Guoliang Xue
Arizona State University
xue@asu.edu

***Abstract*— Recently Choi *et al.* designed the first practical wireless full-duplex system, which challenges the basic assumption in wireless communications that a radio cannot transmit and receive on the same frequency at the same time. Along this line, in this paper we study the cross-layer optimization for routing in full-duplex wireless networks, comprehensively considering various resource competitions and constraints. We first propose a collision-free full-duplex broadcast MAC and prove its necessary and sufficient conditions. We then focus on 1) the problem of how to choose routes to maximize the total profit of multiple users subject to node constraints, and 2) the problem of how to choose routes to minimize the network power consumption subject to the minimum user rate demands and node constraints. We formulate these two problems as convex programming systems. By combining Lagrangian decomposition and subgradient methods, we present distributed iterative algorithms to solve these two problems, which compute the optimized user information flow (i.e. user behavior) on the network layer and the optimized node broadcast rate (i.e. node behavior) on the MAC layer. Our algorithms allow each user and each node to adjust its own behavior individually in each iteration. We prove the convergence, and provide bounds on the amount of constraint violation, and the gap between the optimal solution and our solution in each iteration. Our work comprehensively considers various resource competitions and constraints, and provides a theoretical foundation for the future study on full-duplex wireless networks. To the best of our knowledge, this is the first work to study cross-layer optimization for full-duplex wireless networks.**

***Keywords*-Full-duplex wireless networks; full-duplex broadcast MAC; multipath routing; cross-layer optimization; network power; network utilization; load balance.**

I. Introduction

A basic assumption in wireless communications is that a radio cannot transmit and receive on the same frequency at the same time. A recent research breakthrough challenges this assumption. Choi *et al.* [9] combined antenna, RF, and digital interference cancellation technologies, and designed the first practical single channel wireless full-duplex system. Jain *et al.* [22] then presented a full-duplex radio design using signal inversion and adaptive cancellation. This new design, unlike prior work [9], supports wideband and high power systems. In theory, this new design has no limitation on bandwidth or power. *Therefore, building full-duplex wireless networks (e.g. full-duplex 802.11n wireless networks) becomes possible [22].*

This encouraging breakthrough calls for new research efforts in the design of higher layer protocols and algorithms for wireless networks, and motivates us to re-think of the network design from the theoretical perspective, relaxing this basic assumption, which has been used in wireless communication research during the last decades. In this paper, we concentrate on the cross-layer optimization for routing in full-duplex wireless networks, comprehensively considering various resource competitions and constraints. We study the optimization taking into account the following three network entity roles: *users*, *nodes*, and *network* itself.

From **the perspective of users**, each user obtains an amount of *utility* if a certain information rate is allocated to it. Since the intermediate nodes provide forwarding service for the user, they may charge the user a service fee (i.e. cost). Therefore, each user tries to maximize its *profit* (i.e. utility minus cost) by choosing an optimized *user behavior* (i.e. a route). Furthermore, when there exists more than one user in a wireless network, we need to consider how to solve the resource competition among users, in order to maximize the total user profit, since individual user behaviors usually may not lead to a global optimum.

From **the perspective of nodes**, each node may also have its own *node constraints*, and hence have its own *node behavior*. In this paper, we consider the following two node constraints, which characterize a node individual requirement and a social requirement, respectively. The first node constraint is the *node max load constraint*, more specifically, the maximum load this node is willing to carry. This constraint is of importance since each node may have its own energy consumption concern, or computation and transmission capacity limits. The second node constraint is *the node load balance constraint*. Load balancing is an important issue in the network design [17, 32, 35, 36, 45]. Without considering the node load balance, the traditional routing design methodology that the shortest route is always chosen could result in congestion on the center of a network or hotspots which drain the energy of the nodes in these areas much faster [26, 32]. Security may also be an issue without taking into account the node load balance [32]. For instance, if a large number of messages go through a small number of nodes, then radio jamming can be a vicious attack. In contrast, it would be less effective and more expensive to jam a large number of nodes. Therefore, a node may set a maximum allowed load difference compared with other nodes (such as k-hop neighbors). This constraint can guarantee that a node will not carry too much more load than other nodes, and hence smooths the loads among different nodes. In summary, the node max load constraint reflects a node's own load requirement, and the node load balance constraint shapes the load relationships among different nodes.

From **the perspective of a network**, network energy efficiency is of great importance, since usually wireless nodes are energy-constrained devices, and in some scenarios it seems even infeasible to recharge or replace wireless nodes. These make it significant to develop energy-efficient routing to maximize the network lifetime or minimize the network power consumption subject to the user minimum rate demands.

We hence study the following two problems in this paper.

1) *User profit optimization problem* (UPOP): how to optimize the total profit of multiple simultaneous users in a full-duplex wireless network using multipath routing

This research was supported in part by ARO grant W911NF-09-1-0467 and NSF grants 0830739 and 0905603. The information reported here does not reflect the position or the policy of the federal government.

978-0-7695-4469-4/11 $26.00 © 2011 IEEE
DOI 10.1109/MASS.2011.23

IEEE computer society

subject to node constraints.

2) *Network power consumption optimization problem* (NPCOP): how to optimize the network power consumption in a full-duplex wireless network using multipath routing subject to the minimum user rate demands and node constraints.

In this paper, we realize multipath routing using opportunistic routing (OR), more specifically, the MORE proposed by Chachulski *et al.* in [5]. OR has proved useful and effective for improving the performance of wireless networks by exploiting opportunistic forwarding capacity of intermediate nodes, which overhear packet transmissions [3, 5]. Therefore, OR allows any node overhearing a packet to participate in forwarding it instead of deterministically choosing the next hop before transmitting a packet. Such OR-based multipath routing is called *opportunistic multipath routing (OMR)*.

Our contribution is as follows: We first propose a collision-free full-duplex broadcast MAC and prove its necessary and sufficient conditions. We then formulate UPOP and NPCOP as convex programming systems and present two distributed iterative algorithms to solve UPOP and NPCOP, respectively. Our algorithms compute the optimized user information flow (i.e. user behavior) on the network layer and the optimized node broadcast rate (i.e. node behavior) on the MAC layer. In each iteration, Lagrange multipliers are updated in a distributed manner according to the current user/node behaviors, and then each user and each node individually adjusts its own behavior based on the updated Lagrange multipliers. We prove the convergence of our algorithms, and provide bounds on the amount of feasibility violation and the gap between our solution and the optimal solution in each iteration.

Our work provides a theoretical foundation for the future study on full-duplex wireless networks. To the best of our knowledge, this is the first work to study cross-layer optimization for full-duplex wireless networks. Although we mainly focus on the case where all the nodes are equipped with full-duplex transceivers, in Section VI we will discuss how to extend our work to the case where some nodes can only use traditional half-duplex transceivers.

The rest of this paper is organized as follows. Section II reviews the related work. Section III describes the basic operations of the OMR protocols we consider in this paper. Section IV describes the system model. Section V formulates the problems we are studying, and presents the distributed algorithms to solve these two problems. Section VI discusses the practical issues. Section VII reveals our numerical results. We conclude this paper in Section VIII.

II. Related Works

A. Full-duplex transmission

The current solution to the full-duplex transmission is designed based on the interference cancellation technology. Digital cancellation has been extensively used in the literature, such as ZigZag [18] and successive interference cancellation [20]. Radunović *et al.* [37] suggested RF interference cancellation. Choi *et al.* [9] combined antenna, RF and digital interference cancellation technologies, and designed the first practical single channel wireless full-duplex system. This line of research is further advanced by [11, 12, 39]. Recently, Jain *et al.* [22] presented a full-duplex radio design using signal inversion and adaptive cancellation. This new design, unlike [9], supports wideband and high power systems. In theory, this new design has no limitation on bandwidth or power. Therefore, it is possible to build full-duplex 802.11n devices.

Inspired by these works, full-duplexing has also been applied to other scenarios, such as cognitive radio networks [8] and non-invasive security for implanted medical devices [19].

Our work advances this research progress by studying and optimizing the performance of the higher layers of the wireless networks, which exploit full-duplex transmissions.

B. Opportunistic multipath routing

Opportunistic routing, as a new technology for improving the performance of wireless networks, actually exploits the spatial diversity and broadcast nature of wireless medium, and thus multiple paths are actually used. Biswas and Morris [3] designed ExOR, and Chachulski *et al.* [5] introduced MORE by combining OR and network coding together.

C. Network optimization

We first review the utility or profit maximization problem in networks. Lin and Shroff studied utility maximization problems for communication networks in [28]. Palomar and Chiang [34] presented a systematic framework for network utility maximization to obtain distributed algorithms. Wu *et al.* [43] explored the optimality of utility-based routing through OR without allowing retransmission, and proposed a heuristic solution. Zhang and Li [44] introduced a game theory framework Dice to find a Nash bargaining point of user rates for wireless multipath network coding. Fang *et al.* [14] studied the problem of optimizing network utility or profit for opportunistic routing, taking into account node constraints.

Load balancing is also an important issue in the network design. Gao and Zhang [16] studied wireless network routing algorithms that use only short paths, for minimizing latency, and achieve good load balance, for balancing the energy use. Pham and Perreau [35] showed that multi-path routing provides better congestion and traffic balancing. Further work Ganjali and Keshavarzian [17] showed that multi-path routing can balance load only if a very large number of paths are used. Popa *et al.* [36] showed that an optimum routing scheme based on the shortest paths can be computed by using linear programming. Zorzi and Rao [45] tried to solve the energy-efficiency issues by balancing the load reactively.

Energy efficiency optimization is also an important research topic. Li *et al.* [27] addressed the problem of energy efficient reliable routing for wireless ad hoc networks in the presence of unreliable communication links or devices or lossy wireless link layers by integrating the power control techniques into the energy efficient routing. Mao *et al.* [31] studied energy efficient opportunistic routing in wireless sensor networks and investigated how to select and prioritize forwarder list to minimize energy consumptions. Singh *et al.* [41] presented a case for using new power-aware metrics for determining routes in wireless ad hoc networks. Chang and Tassiulas used linear programming to capture the issue of power consumption in [6], and proposed a centralized algorithm to determine the maximum lifetime in [7]. Sankar and Liu [40] studied the routing problem with the goal of maximizing the network lifetime. Rao *et al.* [38] studied the tradeoff between energy consumption and network performance in real-time wireless sensor networks by investigating the interaction between the network performance optimization and network lifetime maximization problems. Liu *et al.* [29] proposed for sensor networks an energy-aware routing protocol, which improves lifetime by minimizing energy consumption for in-network communications and balancing the load among all the nodes.

The difference between our work and the above line of research is that we study the multipath routing problem considering joint optimization for users, nodes and network in full-duplex wireless networks, and furthermore present distributed optimization algorithms with provable optimality and convergence. To the best of our knowledge, this is the first work to study cross-layer optimization for full-duplex wireless networks.

III. Preliminaries

In this section, we review MORE [5]. The basic operation of our OMR is based on MORE. The difference is that we use a different algorithm to compute routes and node MAC rates.
Source node: Its traffic is divided into a number of batches each with M packets (called *native packets*) in it. The source continuously generates coded packets from each batch using a random linear network coding. Specifically, each coded packet is $m_j^{'} = \sum_i c_{ji} m_i$, where c_{ji} is a random coefficient, and m_i is a native packet from the current batch. $\{c_{j1}, ..., c_{ji}, ..., c_{jM}\}$ is called the *code vector* of packet $m_j^{'}$. The source node keeps generating and sending coded packets from a batch until it receives from the destination the ACK for this batch.
Intermediate node: The sender includes in the *forwarder list* the nodes which are closer (in ETX metric [10]) to the destination than itself, ordered according to their proximity to the destination. Whenever an intermediate node hears a packet, it checks whether it is a forwarder by looking for its ID in the forwarder list. If it is a forwarder for this packet, it then checks whether the packet contains new information (i.e. *an innovative packet*) using simple algebra, such as Gaussian Elimination. A packet is innovative if it is linearly independent from the packets the node has received from this batch. If this packet is innovative, this node generates a random linear combination of the coded packets it has received from the same batch, and broadcasts it. Suppose, for example, that an intermediate node has coded packets of the form $m_j^{'} = \sum_i c_{ji} m_i$. It generates more coded packets by computing a linear combination of these coded packets as follows: $m^{''} = \sum_j a_j m_j^{'}$, where a_j's are random numbers. Obviously, $m^{''}$ is also a linear combination of the native packets (i.e. $m^{''} = \sum_i (\sum_j a_j c_{ji}) m_i$). Since wireless channel is unreliable, MORE uses a heuristic algorithm to compute the expected number of transmissions an intermediate node must make once it receives one innovative packet. *Compared with this heuristic algorithm, in order to optimize the resource allocation, our algorithms compute the optimized user information flow (i.e. user behavior) on the network layer and the optimized node broadcast rate (i.e. node behavior) on the MAC layer.*
Destination node: It keeps all the received innovative packets until M innovative packets from the current batch are received. It then decodes the original whole batch using matrix inversion and sends an ACK back to the source node.

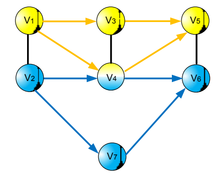


Fig. 1. An example network: if a node is in another node's transmission (or interference) range, they have an edge between them. There are two users in this network. User 1 transmits data from v_1 to v_5 through intermediate nodes v_3 and v_4, and user 2 transmits data from v_2 to v_6 through v_4 and v_7.

IV. System Model

We consider a full-duplex wireless network where there are K simultaneous users. Each user maintains a session from a source node to a destination node and uses intermediate nodes to forward the packets. Our work is *particularly* suitable for the applications with long user sessions (e.g. large file transfer) in static or slowly changing wireless networks (e.g. [1, 2]).

A. Opportunistic multipath routing sub-model

In this subsection, we model the OMR in full-duplex wireless networks. We assume that all the nodes are equipped with single channel full-duplex transceivers. Thus, compared with the node equipped with a traditional half-duplex transceiver, a node in our model is capable of transmitting and receiving on the same frequency simultaneously.

A wireless network is modeled as a directed graph $G = (V, E)$, where E is the set of edges and V is the set of vertices. We use the following terms interchangeably: edge and link, vertex and node. Unlike the traditional unit-disk model, the *transmission range* is defined as the distance where the reception probability falls below a small threshold, as in [44]. There exists a directed edge (u, v) in G if v is in u's transmission range. Since the transmission range is defined as the distance where the reception probability falls below a small threshold, as in [44] we fairly assume that the *interference range* equals to the transmission range. As in [5], for each user a distributed node pre-selection procedure is performed to add intermediate nodes into its forwarder list so that each forwarder is closer (in ETX metric [10]) to the destination than its predecessors. Let $G(V_k, E_k)$ denote the resulting topology involved in the session of user k, where V_k and E_k are the set of nodes and the set of directed links, respectively.

1) Collision-free full-duplex broadcast MAC: We consider a *collision-free full-duplex broadcast MAC* based on slotted scheduling: a broadcast transmission from node u is collision-free if and only if all the other transmitters, which are transmitting packets, are outside the range of any intended downstream receiver of node u. Note that a collision-free broadcast MAC but only supporting half-duplexing was used in [44]. Consider the example shown in Fig.1, where v_3 and v_4 are the intended downstream receivers of node v_1. If no collision occurs on v_3, the transmitters of nodes v_4 and v_5 should not transmit, and if no collision occurs on v_4, the transmitters of nodes v_2, v_3, v_5 and v_6 should not transmit. Thus a collision-free broadcast transmission from node v_1 means that no collision occurs on both v_3 and v_4. Since full-duplexing is supported, each node is able to receive packets while transmitting. For example, v_7 can transmit the previously received packets to v_6 while receiving packets from v_2. Note that although v_3 (or v_4) is also capable of transmitting and receiving simultaneously, when it is receiving packets from v_1, it should not transmit because that transmission will interfere with v_4 (or v_3). Although we focus on this MAC model, in Section VI we will discuss how to integrate our work into other possible full-duplex MACs.

Let $I(u)$ denote the set of nodes whose transmission (interference) range u is located in. For example, in the network shown in Fig.1, $I(v_1)$=$\{v_2, v_3, v_4\}$, $I(v_2)$=$\{v_1, v_4, v_7\}$, and $I(v_4)$=$\{v_1, v_2, v_3, v_5, v_6\}$. We use $Z_k^{(t)}(u)$ to denote a binary variable indicating whether node u transmits user k's data in slot t. We assume that a node can only transmit one user's data in one time slot. This assumption is reasonable since the lengths of time slots can be designed to be short enough so that a node only transmits one user's data in one time slot. *A necessary and sufficient condition for collision-free broadcasts* is that *for any* $k \in [1, K]$*, the following inequality holds*:

$$\sum_{m\in[1,K]} \sum_{v\in I(u)} Z_m^{(t)}(v) \leq 1, \forall u \in V_k \setminus s_k. \quad (1)$$

It means that any node u allows the broadcast transmission from at most one transmitter within its range (excluding its transmit antenna). For user k its source s_k is excluded, since s_k does not need to receive data from other nodes. Before we prove this necessary and sufficient condition, let us use the example shown in Fig.1 for an illustration. Assume that user 1 transmits packets from v_1 to v_5 through v_3 and v_4, and that user 2 transmits packets from v_2 to v_6 through v_4 and v_7. Thus $V_1 = \{v_1, v_3, v_4, v_5\}$ and $V_2 = \{v_2, v_4, v_6, v_7\}$. The necessary and sufficient condition can be expressed as:

for $k = 1$,

$$\text{for } v_3, \quad Z_1^{(t)}(v_1) + Z_1^{(t)}(v_4) + Z_2^{(t)}(v_4) \leq 1; \quad (2)$$

$$\text{for } v_4, \quad Z_1^{(t)}(v_1) + Z_1^{(t)}(v_3) + Z_2^{(t)}(v_2) \leq 1; \quad (3)$$

$$\text{for } v_5, \quad Z_1^{(t)}(v_3) + Z_1^{(t)}(v_4) + Z_2^{(t)}(v_4) \leq 1; \quad (4)$$

and for $k = 2$,

$$\text{for } v_4, \quad Z_1^{(t)}(v_1) + Z_1^{(t)}(v_3) + Z_2^{(t)}(v_2) \leq 1; \quad (5)$$

$$\text{for } v_6, \quad Z_1^{(t)}(v_4) + Z_2^{(t)}(v_4) + Z_2^{(t)}(v_7) \leq 1; \quad (6)$$

$$\text{for } v_7, \quad Z_2^{(t)}(v_2) \leq 1, \quad (7)$$

where $Z_k^{(t)}(u) \in \{0, 1\}$ for $k = 1$ or 2 and $u \in V$. Note that $Z_1^{(t)}(v_2) = Z_1^{(t)}(v_5) = Z_1^{(t)}(v_6) = Z_1^{(t)}(v_7) = Z_2^{(t)}(v_1) = Z_2^{(t)}(v_3) = Z_2^{(t)}(v_5) = Z_2^{(t)}(v_6) = 0$. (2) means that if v_3 can receiver packets of user 1 from v_1 without collision, v_4 cannot transmit. Note that since full-duplexing is used, v_3 does not have to care whether it is transmitting. Although (3) and (5) are the same, they convey different meanings. (3) means that if v_4 can receiver packets of user 1 from v_1 without collision, v_2 and v_3 cannot transmit. (5) means that if v_4 can receiver packets of user 2 from v_2 without collision, v_1 and v_3 cannot transmit. Likewise, since full-duplexing is used, v_4 does not have to care whether it is transmitting. Note that although (3) and (5) convey different meanings, in real computation, *such redundant inequalities should be removed to reduce computational complexity*. In addition, we do not have the inequalities for v_1 and v_2 since they are sources and do not need to receive data from other nodes.

We now prove this necessary and sufficient condition. On one hand, if for any $k \in [1, K]$ inequality (1) holds, for any node *at most* one of its intended upstream transmitters is broadcasting. Due to the full-duplex operation, no matter whether this node is transmitting, it can hear this transmission without collision. On the other hand, if condition (1) does not hold for some $k \in [1, K]$ and some node $u \in V_k \setminus s_k$, then $\sum_{m\in[1,K]} \sum_{v\in I(u)} Z_m^{(t)}(v) \geq 2$. Note that a node can only transmit one user's data in one time slot. This means that at least two of u's upstream transmitters are broadcasting, and as a result a collision occurs.

Assuming the period of a schedule is T, for any $k \in [1, K]$ we thus have

$$\frac{\mathbf{C}}{T} \sum_{t\in[1,T]} \sum_{m\in[1,K]} \sum_{v\in I(u)} Z_m^{(t)}(v) \leq \mathbf{C}, \forall u \in V_k \setminus s_k,$$

where $\mathbf{C}$ is the MAC layer capacity, which is the maximum broadcast rate of a node when no interferer presents. Since the *average broadcast rate* of node u for user m (i.e. ***node*** $\boldsymbol{u}$***'s behavior for user*** $\boldsymbol{m}$) can be computed as $b_m(u) = \lim_{T\to\infty} \frac{\mathbf{C}}{T} \sum_{t\in[1,T]} Z_m^{(t)}(u)$, we must have for any $k \in [1, K]$

$$\sum_{m\in[1,K]} \sum_{v\in I(u)} b_m(v) \leq \mathbf{C}, \forall u \in V_k \setminus s_k. \quad (8)$$

Note that since we transformed an integer variable $Z_m^{(t)}(u)$ into a continuous one $b_m(u)$ by averaging, constraint (8) is necessary but not necessarily sufficient.

Remark: We note that full-duplex transmission could lead to other routing designs such as wormhole routing [9]. As the first attempt towards the routing optimization for full-duplex wireless networks, this paper mainly focuses on the routing optimization based on this collision-free broadcast MAC.

2) Information flow: Considering the flow conservation principle, for any $u \in V$ we have

$$\sum_{(u,v)\in E_k} r_k(u, v) - \sum_{(w,u)\in E_k} r_k(w, u) = \Delta_k(u), \forall k \in [1, K], \quad (9)$$

where $\Delta_k(u)$ is equal to R_k if $u = s_k$, $-R_k$ if $u = d_k$, and 0 otherwise. s_k and d_k denote user k's source and destination, respectively. R_k and $r_k(u, v)$ denote user k's *total information rate* and *link information rate* on link (u, v), respectively. The link information rate vector $\overrightarrow{r_k}$ represents ***user k's behavior*** (i.e. ***user k's route***).

3) Network coding constraint: Our OMR is realized by using network coding based OR (see the operations of MORE in Section III). Although Lun *et al.* [30] made an exact characterization of the coding model, due to an exponential number of constraints, it makes the problem intractable. To balance the model accuracy and the computation tractability, we adopt the following model which was proposed in [44]:

$$b_k(u) \cdot \boldsymbol{p}(u, v) \geq r_k(u, v), \forall (u, v) \in E_k, \quad (10)$$

where $\boldsymbol{p}(u, v)$>0 is the packet delivery ratio of link (u, v). Note that this is not a tight bound. However, as discussed in [44], this is an effective approximation to the behavior of an actual wireless network using network coding. It includes the tractable information that users and nodes can exploit to induce a better performance.

B. User profit sub-model

User k obtains a utility of $\mathcal{U}_k(R_k)$ if it achieves a total information rate of R_k. We assume that $\mathcal{U}_k(\cdot)$ is an increasing, concave and continually differentiable function as in [25]. Since intermediate nodes provide forwarding service, each forwarder charges this user a service fee which is proportional to the service rate (i.e. the broadcast rate) for this user. Let $\boldsymbol{\lambda}_k(u)$ denote the service price charged by node u for user k. User k hence pays $\boldsymbol{\lambda}_k(u)b_k(u)$ ($\boldsymbol{\lambda}_k(u) \geq 0$) for node u's forwarding service. User k's profit is thus defined by

$$\mathcal{P}_k(R_k, \overrightarrow{b}_k) = \mathcal{U}_k(R_k) \quad - \sum_{u \in V_k \setminus d_k} \boldsymbol{\lambda}_k(u) b_k(u). \tag{11}$$

C. Node constraint sub-model

1) Node max load constraint: Due to the energy consumption concern, or computation and transmission capacity limits, each node u needs to consider the maximum average broadcast rate $\boldsymbol{\varphi}(u)$ it is willing to provide. As a result, node u sets an upper bound on its average broadcast rate:

$$b(u) = \sum_{k \in [1,K]} b_k(u) \leq \boldsymbol{\varphi}(u), \boldsymbol{\varphi}(u) > 0. \tag{12}$$

2) Node load balance constraint: We define a *load balance area* (i.e. a set of nodes) for each node. We use $v \in L(u)$ to represent that v is in u's load balance area. That is to say,

$$|b(u) - b(v)| \leq \boldsymbol{\theta}(u,v), \boldsymbol{\theta}(u,v) > 0, \forall v \in L(u), \tag{13}$$

where $\boldsymbol{\theta}(u,v)$ is a parameter set by node u. $L(u)$ and $\boldsymbol{\theta}(u,v)$ are used by node u to achieve a controllable balanced load with other nodes. In this paper, we consider a symmetric load balance (i.e. $v \in L(u) \Longleftrightarrow u \in L(v)$, and $\boldsymbol{\theta}(u,v)=\boldsymbol{\theta}(v,u)$). Although in practice the load balance constraint could be asymmetric, there would be no big difference for our mathematical analysis and algorithm. We hence concentrate on this simpler definition which makes our expressions clearer.

D. Network power consumption sub-model

Based on our slotted MAC, we assume that a node's power consumption is proportional to its broadcast rate, since in most cases transmission operations dominate the energy consumption [42]. Specifically, node u's average power consumption is computed as $\boldsymbol{c}(u)b(u)$, where $\boldsymbol{c}(u)$ is a *power consumption ratio*. The network power consumption is computed as:

$$\mathcal{N}(\overrightarrow{b}) = \sum_{k \in [1,K]} \sum_{u \in V_k \setminus d_k} \boldsymbol{c}(u) b_k(u). \tag{14}$$

V. Problem Formulation and Distributed Optimization Algorithms

In this section, we first formulate UPOP and NPCOP as two convex programming systems, then present distributed algorithms to solve these two systems, and finally analyze the convergence and the optimality of our algorithms.

A. Problem formulation

1) User profit optimization problem: By our user profit submodel (11), OMR submodel (8)-(10), and node constraint submodel (12)(13), UPOP is formulated as:

$$\begin{aligned}
&System\ 1(\overrightarrow{r}, \overrightarrow{b}) : \max \sum_{k \in [1,K]} \Big(\mathcal{U}_k(R_k) - \sum_{u \in V_k \setminus d_k} \boldsymbol{\lambda}_k(u) b_k(u) \Big), \\
&\text{s.t.} \sum_{(u,v) \in E_k} r_k(u,v) - \sum_{(w,u) \in E_k} r_k(w,u) = \Delta_k(u), \\
&\qquad \forall u \in V, k \in [1,K]; \qquad (1) \\
&\sum_{m \in [1,K]} \sum_{v \in I(u)} b_m(v) \leq \mathbf{C}, \forall u \in V_k \setminus s_k, k \in [1,K]; \qquad (2) \\
&b_k(u) \cdot \boldsymbol{p}(u,v) \geq r_k(u,v), \forall (u,v) \in E_k, k \in [1,K]; \qquad (3) \\
&\sum_{k \in [1,K]} b_k(u) \leq \boldsymbol{\varphi}(u), \forall u \in V; \qquad (4) \\
&|\sum_{k \in [1,K]} b_k(u_1) - \sum_{k \in [1,K]} b_k(u_2)| \leq \boldsymbol{\theta}(u_1,u_2), \\
&\forall u_2 \in L(u_1), u_1 \in V; \qquad (5) \\
&\text{over} \;\; r_k(u,v) \in [0,\mathbf{C}], \forall (u,v) \in E_k, k \in [1,K], \\
&\qquad b_k(u) \in [0,\mathbf{C}], \forall u \in V_k \setminus d_k, k \in [1,K].
\end{aligned}$$

2) Network power consumption optimization problem: User k has a *minimum user rate demand* $\boldsymbol{\Lambda}_k$. By our network power consumption submodel (14), OMR submodel (8)-(10), node constraint submodel (12)(13), and minimum user rate demand, NPCOP is formulated as System 2. We will discuss how to select $\boldsymbol{\Lambda}_k$ in Section VI, since if some of them are too large, the network resource may not be able to satisfy them and as a result System 2 may not have a feasible solution.

$$\begin{aligned}
&System2(\overrightarrow{r}, \overrightarrow{b}) : \min \sum_{k \in [1,K]} \sum_{u \in V_k \setminus d_k} \boldsymbol{c}(u) b_k(u), \\
&\text{s.t.} \ (1) - (5); \\
&\quad R_k \geq \boldsymbol{\Lambda}_k, k \in [1,K]; \qquad (6) \\
&\quad \text{over} \;\; r_k(u,v) \in [0,\mathbf{C}], \forall (u,v) \in E_k, k \in [1,K], \\
&\qquad b_k(u) \in [0,\mathbf{C}], \forall u \in V_k \setminus d_k, k \in [1,K].
\end{aligned}$$

Remark: These two systems are independent, which target different optimization objectives.

B. Distributed optimization algorithms: Pathbook

Although Systems 1 and 2 can be solved using traditional convex programming techniques [4], distributed algorithms are preferable for the purpose of practical implementations. We combine Lagrangian decomposition with the approximate dual subgradient method (ADSM) proposed by Nedić and Ozdaglar in [33], and hence propose distributed algorithms for solving Systems 1 and 2 (denoted by *Pathbook-I* and *Pathbook-II*, respectively). ADSM can be summarized as follows.

The ***primal problem*** is the following:

$$\min \ f(\overrightarrow{x}), \qquad \text{s.t.} \ g(\overrightarrow{x}) \preceq 0, \qquad \text{over} \ \overrightarrow{x} \in \overrightarrow{X}, \tag{7}$$

where $f(\cdot) : \mathbb{R}^N \mapsto \mathbb{R}$ is a convex function ($\mathbb{R}^N$ is an N-dimensional vector space), $g(\cdot) = (g_1(\cdot), ..., g_\rho(\cdot))^T$ and each $g_j(\cdot) : \mathbb{R}^N \mapsto \mathbb{R}$ is a convex function, and $\overrightarrow{X} \in \mathbb{R}^N$ is a nonempty compact convex set. $(\cdot)^T$ denotes the transpose of $(\cdot)$. The ***dual problem*** is the following:

$$\max \ q(\overrightarrow{\delta}) = \inf_{\overrightarrow{x} \in \overrightarrow{X}} \big(L(\overrightarrow{x}, \overrightarrow{\delta}) \big), \quad \text{s.t.} \ \overrightarrow{\delta} \succeq 0, \quad \text{over} \ \overrightarrow{\delta} \in \mathbb{R}^\rho,$$

where $L(\overrightarrow{x}, \overrightarrow{\delta}) = f(\overrightarrow{x}) + \overrightarrow{\delta}^T g(\overrightarrow{x})$ is the Lagrangian of the primal problem, and $\overrightarrow{\delta}$ is the vector of Lagrange multipliers.

ADSM iteratively computes $\overrightarrow{\delta}^{(1)}, \overrightarrow{x}^{(1)}, \overrightarrow{\delta}^{(2)}, \overrightarrow{x}^{(2)}, \cdots$ from a starting point $\{\overrightarrow{\delta}^{(0)}, \overrightarrow{x}^{(0)}\} \in \mathbb{R}^\rho \times \overrightarrow{X}$ as follows:

$$\begin{aligned}
&\text{for } i \geq 0, \overrightarrow{\delta}^{(i+1)} = [\overrightarrow{\delta}^{(i)} + \eta g(\overrightarrow{x}^{(i)})]^+, \text{where } \overrightarrow{x}^{(i)} \in \\
&\text{argmin}_{\overrightarrow{x} \in \overrightarrow{X}} \Big(f(\overrightarrow{x}) + (\overrightarrow{\delta}^{(i)})^T g(\overrightarrow{x}) \Big), \qquad (8)
\end{aligned}$$

where η is a constant stepsize. For a vector $\overrightarrow{x} \in \mathbb{R}^N$, $\overrightarrow{x}^+$ is the projection of $\overrightarrow{x}$ on the nonnegative orthant in $\mathbb{R}^N$, i.e. $\overrightarrow{x}^+ = (\max\{0, x_1\}, ..., \max\{0, x_N\})^T$ for $\overrightarrow{x} = (x_1, ..., x_N)^T$. The primal solution is approximated using averaging:

$$\hat{x}^{(i)} = \frac{1}{i}\sum_{m=0}^{i-1}\overrightarrow{x}^{(m)}, i \geq 1. \quad (9)$$

Our distributed algorithms use subgradient methods in dual space and exploit the subgradient information to produce primal near-feasible and near-optimal solutions. The high-level frameworks of Pathbook-I and Pathbook-II are based on ADSM, and thus are similar. We now briefly outline this framework. Let $\overrightarrow{\delta}^{(i)}$ denote the vector of Lagrange multipliers in the i^{th} iteration, and $\overrightarrow{x}^{(i)}$ denote the vector $((\overrightarrow{r}^{(i)})^T, (\overrightarrow{b}^{(i)})^T)^T$. According to ADSM, from a starting point we compute $\overrightarrow{\delta}^{(1)}$, and based on $\overrightarrow{\delta}^{(1)}$, $\overrightarrow{x}^{(1)}$ can be computed. In the second iteration, we compute $\overrightarrow{\delta}^{(2)}$ based on $\overrightarrow{\delta}^{(1)}$ and $\overrightarrow{x}^{(1)}$, and based on $\overrightarrow{\delta}^{(2)}$, $\overrightarrow{x}^{(2)}$ can be computed. This procedure continues until a stop criteria is reached. Using Lagrangian decomposition, the update operations for both $\overrightarrow{\delta}^{(i)}$ and $\overrightarrow{x}^{(i)}$ can be done in a distributed manner.

*1) **Distributed optimization algorithm Pathbook-I for UPOP**:* Due to our symmetric definition of load balance areas, the absolute value sign in (5) can be omitted. After some simple mathematical manipulations and removing redundant inequalities, the Lagrangian of System 1 subject to constraint (1) can be expressed as:

$$\begin{aligned}
&L(\overrightarrow{r}, \overrightarrow{b}, \overrightarrow{\alpha}, \overrightarrow{\beta}, \overrightarrow{\mu}, \overrightarrow{\omega}) \\
&= -\sum_{k\in[1,K]}\mathcal{U}_k(R_k) + \sum_{k\in[1,K]}\sum_{u\in V}\boldsymbol{\lambda}_k(u)b_k(u) \\
&\quad + \sum_{u\in V}\alpha(u)\big(\sum_{k\in[1,K]}\sum_{v\in I(u),u\neq s_k} b_k(v) - \mathbf{C}\big) \\
&\quad + \sum_{k\in[1,K]}\sum_{(u,v)\in E_k}\beta_k(u,v)\big(r_k(u,v) - b_k(u)\cdot \boldsymbol{p}(u,v)\big) \\
&\quad + \sum_{u\in V}\mu(u)\big(\sum_{k\in[1,K]} b_k(u) - \boldsymbol{\varphi}(u)\big) \\
&\quad + \sum_{u_1,u_2\in V:u_2\in L(u_1)}\omega(u_1,u_2)\Big(\sum_{k\in[1,K]}\big(b_k(u_1) - b_k(u_2)\big) - \boldsymbol{\theta}(u_1,u_2)\Big). \quad (10)
\end{aligned}$$

In the i^{th} iteration, according to (8) we update Lagrange multipliers as follows:

$$\begin{aligned}
&\forall u \in V, \alpha^{(i)}(u) = [\alpha^{(i-1)}(u) + \eta A_u^{(i-1)}]^+, \\
&\forall (u,v) \in E_k, \beta_k^{(i)}(u,v) = [\beta_k^{(i-1)}(u,v) + \eta B_{(k,u,v)}^{(i-1)}]^+, \\
&\forall u \in V, \mu^{(i)}(u) = [\mu^{(i-1)}(u) + \eta M_u^{(i-1)}]^+, \\
&\forall u \in V \text{and } v \in L(u), \omega^{(i)}(u,v) = [\omega^{(i-1)}(u,v) + \eta \Omega_{(u,v)}^{(i-1)}]^+,
\end{aligned}$$

where

$$\begin{aligned}
A_u^{(i-1)} &= \sum_{k\in[1,K]}\sum_{v\in I(u),u\neq s_k} b_k^{(i-1)}(v) - \mathbf{C}, \\
B_{(k,u,v)}^{(i-1)} &= r_k^{(i-1)}(u,v) - b_k^{(i-1)}(u)\cdot \boldsymbol{p}(u,v), \\
M_u^{(i-1)} &= \sum_{k\in[1,K]} b_k^{(i-1)}(u) - \boldsymbol{\varphi}(u), \\
\Omega_{(u,v)}^{(i-1)} &= \sum_{k\in[1,K]}\big(b_k^{(i-1)}(u) - b_k^{(i-1)}(v)\big) - \boldsymbol{\theta}(u,v).
\end{aligned}$$

After Lagrange multipliers are computed, according to (8), we need to find $\overrightarrow{r}^{(i)}$ and $\overrightarrow{b}^{(i)}$ so as to minimize $L(\overrightarrow{r}, \overrightarrow{b}, \overrightarrow{\alpha}^{(i)}, \overrightarrow{\beta}^{(i)}, \overrightarrow{\mu}^{(i)}, \overrightarrow{\omega}^{(i)})$. After some mathematical manipulations, (10) leads to the following:

$$\begin{aligned}
&L(\overrightarrow{r}, \overrightarrow{b}, \overrightarrow{\alpha}, \overrightarrow{\beta}, \overrightarrow{\mu}, \overrightarrow{\omega}) \\
&= -\left(\sum_{k\in[1,K]}\mathcal{U}_k(R_k) + \sum_{k\in[1,K]}\sum_{(u,v)\in E_k}\beta_k(u,v)r_k(u,v)\right) \\
&\quad + \Bigg(\sum_{k\in[1,K]}\sum_{u\in V}\boldsymbol{\lambda}_k(u)b_k(u) + \sum_{k\in[1,K]}\sum_{u\in V_k}\sum_{v\in I(u),v\neq s_k}\alpha(v)b_k(u) \\
&\quad - \sum_{k\in[1,K]}\sum_{(u,v)\in E_k}\beta_k(u,v)b_k(u)\boldsymbol{p}(u,v) + \sum_{u\in V}\mu(u)\sum_{k\in[1,K]} b_k(u) \\
&\quad + \sum_{u_1,u_2\in V:u_2\in L(u_1)}\omega(u_1,u_2)\sum_{k\in[1,K]}\big(b_k(u_1) - b_k(u_2)\big)\Bigg) - \mathbf{C}\sum_{u\in V}\alpha(u) \\
&\quad - \boldsymbol{\theta}(u_1,u_2)\sum_{u_1,u_2\in V:u_2\in L(u_1)}\omega(u_1,u_2) - \sum_{u\in V}\mu(u)\boldsymbol{\varphi}(u). \quad (11)
\end{aligned}$$

Fortunately, by (11) this minimization operation can be decomposed to users' behaviors and nodes' behaviors. Specifically, in the i^{th} iteration user k solves the following problem:

$$\begin{aligned}
\text{min} \quad & -\mathcal{U}_k(R_k^{(i)}) + \sum_{(u,v)\in E_k}\beta_k^{(i)}(u,v)r_k^{(i)}(u,v), \quad (12) \\
\text{s.t.} \quad & \sum_{(u,v)\in E_k} r_k^{(i)}(u,v) - \sum_{(w,u)\in E_k} r_k^{(i)}(w,u) = \Delta_k^{(i)}(u), u \in V_k, \\
\text{over} \quad & r_k^{(i)}(u,v) \in [0,\mathbf{C}], \forall (u,v) \in E_k,
\end{aligned}$$

and node u solves the following problem:

$$\begin{aligned}
\text{min} \quad & \Big(\boldsymbol{\lambda}_k(u) + \sum_{v\in I(u),v\neq s_k}\alpha^{(i)}(v) - \sum_{(u,v)\in E_k}\beta_k^{(i)}(u,v)\boldsymbol{p}(u,v) \\
& +\mu^{(i)}(u) + \sum_{v\in L(u)}\omega^{(i)}(u,v) - \sum_{v\in L(u)}\omega^{(i)}(v,u)\Big)b_k^{(i)}(u), \quad (13) \\
\text{over} \quad & b_k^{(i)}(u) \in [0,\mathbf{C}], u \in V_k \setminus d_k.
\end{aligned}$$

Obviously, (13) can be solved by node u. Now we describe how to solve (12). Based on the flow-path formulation technique in [44], we consider an equivalent problem:

$$\begin{aligned}
\text{min} \quad & -\mathcal{U}_k(\sum_{\pi\in P}\gamma_k(\pi)) + \sum_{\pi\in P}\kappa_k(\pi)\gamma_k(\pi), \\
\text{over} \quad & \sum_{\pi\in P}\gamma_k(\pi) \in [0,\mathbf{C}],
\end{aligned}$$

where P is a set of single paths, $\gamma_k(\pi)$ is the flow rate of user k following path π, and $\kappa_k(\pi) = \sum_{(u,v)\in\pi}\beta_k^{(i)}(u,v)$. Solving this equivalent problem will result in that the min-cost single path (with respect to $\beta_k^{(i)}(u,v)$) is always chosen. We use Γ_k to denote the value of this single path flow, and hence transform the problem above to

$$\text{min } -\mathcal{U}_k(\Gamma_k) + \kappa_k^{min}\Gamma_k, \text{ over } \Gamma_k \in [0,\mathbf{C}], \quad (14)$$

where κ_k^{min} is the cost of the min-cost path (with respect to $\beta_k^{(i)}(u,v)$). Therefore, each user k can use a distributed shortest path algorithm to compute κ_k^{min}, and then solves (14).

$r_k^{(i)}(u,v)$ is set to the solution to (14) if (u,v) is on this min-cost path, and 0 otherwise.

In the i^{th} iteration, according to (9) users and nodes can approximate the primal solutions in the following way:

$$\hat{r}_k^{(i)}(u,v)=\frac{1}{i}\sum_{m=0}^{i-1} r_k^{(m)}(u,v), \hat{b}_k^{(i)}(u)=\frac{1}{i}\sum_{m=0}^{i-1} b_k^{(m)}(u), i \geq 1. \quad (15)$$

After the i^{th} iteration, user k uses $\hat{r}_k^{(i)}(u,v)$ as its user behavior on link (u,v) and node u uses $\hat{b}_k^{(i)}(u)$ as its node behavior for user k until the next iteration. (15) tells each user its optimized information flow. However, transmissions in our OMR are opportunistic. To realize this information flow, we adopt a *transmission credit mechanism* (see Section VI).

We now analyze what information is needed in each iteration. We first analyze the update operation of Lagrange multipliers. For $\alpha^{(i)}(u)$, node u needs its one-hop neighbors' broadcast rates. For $\beta_k^{(i)}(u,v)$, node u needs $b_k^{(i)}(u)$ and $r_k^{(i)}(u,v)$. For $\mu^{(i)}(u)$, node u needs its own broadcast rate $b_k^{(i)}(u)$. For $\omega^{(i)}(u,v)$, u needs the broadcast rates of the nodes located in its load balance area. We then analyze the update operation of user and node behaviors when the Lagrange multipliers have been computed. Each user can update its user behavior by using a distributed shortest path algorithm. For node behaviors, each node needs the information from its one-hop neighbors and the nodes in its load balance area. This analysis indicates that no global information is needed as long as no node includes all the nodes in its load balance area. Note that a large load balance area will introduce many information exchanges. We will discuss this issue in Section VI. This analysis also implies that the iteration starting point $((\overrightarrow{x}^{(0)})^T, (\overrightarrow{\delta}^{(0)})^T)^T$ can be negotiated in a distributed manner.

*2) **Distributed optimization algorithm pathbook-II for NPCOP**:* As in Pathbook-I, we first compute the Lagrangian of System 2 subject to constraint (1) as follows:

$$\begin{aligned}
&L(\overrightarrow{r}, \overrightarrow{b}, \overrightarrow{\alpha}, \overrightarrow{\beta}, \overrightarrow{\mu}, \overrightarrow{\omega}, \overrightarrow{y}) \\
=& \sum_{k\in[1,K]} \sum_{u\in V_k\setminus d_k} \boldsymbol{c}(u) b_k(u) \\
+& \sum_{u\in V} \alpha(u)\Big(\sum_{k\in[1,K]} \sum_{v\in I(u), u\neq s_k} b_k(v) - \mathbf{C}\Big) \\
+& \sum_{k\in[1,K]} \sum_{(u,v)\in E_k} \beta_k(u,v)\big(r_k(u,v) - b_k(u)\boldsymbol{p}(u,v)\big) \\
+& \sum_{u\in V} \mu(u)\Big(\sum_{k\in[1,K]} b_k(u) - \boldsymbol{\varphi}(u)\Big) \\
+& \sum_{u_1,u_2\in V: u_2\in L(u_1)} \omega(u_1,u_2)\Big(\sum_{k\in[1,K]}\big(b_k(u_1) - b_k(u_2)\big) - \boldsymbol{\theta}(u_1,u_2)\Big) \\
+& \sum_{k\in[1,K]} y_k(\boldsymbol{\Lambda}_k - R_k). \qquad (16)
\end{aligned}$$

In the i^{th} iteration, according to (8) we first need to update Lagrange multipliers. For $\alpha^{(i)}(u)$, $\beta_k^{(i)}(u,v)$, $\mu^{(i)}(u)$, and $\omega^{(i)}(u,v)$, we can update them as in Pathbook-I. For $y_k^{(i)}$, we update it as follows: $y_k^{(i)} = [y_k^{(i-1)} + \eta Y_k^{(i-1)}]^+$, where $Y_k^{(i-1)} = \boldsymbol{\Lambda}_k - R_k^{(i-1)}$.

After Lagrange multipliers are computed, according to (8), we need to find $\overrightarrow{r}^{(i)}$ and $\overrightarrow{b}^{(i)}$ so as to minimize $L(\overrightarrow{r}, \overrightarrow{b}, \overrightarrow{\alpha}^{(i)}, \overrightarrow{\beta}^{(i)}, \overrightarrow{\mu}^{(i)}, \overrightarrow{\omega}^{(i)}, \overrightarrow{y}^{(i)})$. As in Pathbook-I, after some manipulations, we decompose this minimization operation to the following users' behaviors and nodes' behaviors.

In the i^{th} iteration, user k solves the following problem:

$$\begin{aligned}
\min \quad & -y_k^{(i)} R_k^{(i)} + \sum_{(u,v)\in E_k} \beta_k^{(i)}(u,v) r_k^{(i)}(u,v), \qquad (17)\\
\text{s.t.} \quad & \sum_{(u,v)\in E_k} r_k^{(i)}(u,v) - \sum_{(w,u)\in E_k} r_k^{(i)}(w,u) = \Delta_k^{(i)}(u), u\in V_k,\\
\text{over} \quad & r_k^{(i)}(u,v)\in[0,\mathbf{C}], \forall(u,v)\in E_k,
\end{aligned}$$

and node u solves the following problem:

$$\begin{aligned}
\min \quad & \Big(\boldsymbol{c}(u) + \sum_{v\in I(u), v\neq s_k} \alpha^{(i)}(v) - \sum_{(u,v)\in E_k} \beta_k^{(i)}(u,v)\boldsymbol{p}(u,v)\\
&+\mu^{(i)}(u) + \sum_{v\in L(u)} \omega^{(i)}(u,v) - \sum_{v\in L(u)} \omega^{(i)}(v,u)\Big) b_k^{(i)}(u), \qquad (18)\\
\text{over} \quad & b_k^{(i)}(u)\in[0,\mathbf{C}], u\in V_k\setminus d_k.
\end{aligned}$$

Problem (18) can be easily solved by node u. For user problem (17), as before we consider an equivalent problem:

$$\begin{aligned}
\min \quad & -y_k^{(i)} \sum_{\pi\in P} \gamma_k(\pi) + \sum_{\pi\in P} \kappa_k(\pi)\gamma_k(\pi),\\
\text{over} \quad & \sum_{\pi\in P} \gamma_k(\pi) \in [0,\mathbf{C}].
\end{aligned}$$

Solving this equivalent problem will result in that the min-cost path is always chosen. Thus this problem is transformed to

$$\min \; -y_k^{(i)}\Gamma_k + \kappa_k^{min}\Gamma_k, \text{ over } \Gamma_k\in[0,\mathbf{C}], \qquad (19)$$

where κ_k^{min} is the cost of the min-cost path (with respect to $\beta_k^{(i)}(u,v)$). Therefore, each user k can use a distributed shortest path algorithm to find the min-cost path (with respect to $\beta_k^{(i)}(u,v)$) so as to compute κ_k^{min}, and then solves (19). $r_k^{(i)}(u,v)$ is set to the solution to (19) if (u,v) is on this min-cost path, and 0 otherwise.

Users and nodes can approximate the primal solutions using (15). After the i^{th} iteration, user k uses $\hat{r}_k^{(i)}(u,v)$ as its user behavior on link (u,v) and node u uses $\hat{b}_k^{(i)}(u)$ as its node behavior for user k until the next iteration. We now analyze what information is needed in each iteration. Similar to UPOP, the update operations for $\overrightarrow{r}^{(i)}$, $\overrightarrow{b}^{(i)}$, $\overrightarrow{\alpha}^{(i)}$, $\overrightarrow{\beta}^{(i)}$, $\overrightarrow{\mu}^{(i)}$, and $\overrightarrow{\omega}^{(i)}$ do not require global information. For $y_k^{(i)}$, obviously it can be updated by user k itself. Thus, no global information is required in the iterations.

C. Optimality and convergence analysis

Theorem 5.1 characterizes the optimality and convergence of our distributed algorithms. Let $\hat{x}^{(i)}$ denote the primal solution approximated in the i^{th} iteration. Note that we reformulate Systems 1 and 2 to the standard form (7) for brevity.

Theorem 5.1: The following properties hold for all $i \geq 1$:

1) An upper bound on the amount of constraint violation of the vector $\hat{x}^{(i)}$ for System 1 (System 2) is given by $||g(\hat{x}^{(i)})^+|| \leq \frac{\mathcal{B}}{i\eta}$, where $\mathcal{B} = \frac{2}{\varpi}(f(\overrightarrow{x^s}) - q(\overrightarrow{\delta}^{(0)})) + \max\{||\overrightarrow{\delta}^{(0)}||, \frac{1}{\varpi}(f(\overrightarrow{x^s}) - q(\overrightarrow{\delta}^{(0)}) + \frac{\eta\mathcal{L}^2}{2}) + \eta\mathcal{L}\}$ ($\overrightarrow{x^s}$ is a Slater vector, $\varpi = \min_{j\in[1,\rho]}(-g_j(\overrightarrow{x^s}))$, and $\mathcal{L} = \max_{\overrightarrow{x}\in\overrightarrow{X}} ||g(\overrightarrow{x})||$). Note that $\varpi>0$ since $\overrightarrow{x^s}$ is a Slater

vector, and that $\mathcal{L}$ is bounded since considering that the domain of each element of $\overrightarrow{x}$ is $[0, \mathbf{C}]$, we know $\overrightarrow{X}$ is a compact set [33].

2) An upper bound on $f(\hat{x}^{(i)})$ is given by $f(\hat{x}^{(i)}) \leq f^* + \frac{||\overrightarrow{\delta}^{(0)}||^2}{2i\eta} + \frac{\eta\mathcal{L}^2}{2}$, where f^* is primal optimal value of System 1(System 2).
3) A lower bound on $f(\hat{x}^{(i)})$ is given by $f(\hat{x}^{(i)}) \geq f^*$- $\frac{\mathcal{B}^2}{i\eta}$.□

The proof is given in Appendix A. Theorem 5.1.1) states that the amount of constraint violation of our primal solution $\hat{x}^{(i)}$ diminishes to zero at the rate $\frac{1}{i}$ as the number of iterations i increases. 2) and 3) imply that as i increases, the primal value of System 1 (System 2) using our solution $\hat{x}^{(i)}$ converges to f^* within error level $\frac{\eta\mathcal{L}^2}{2}$ with the rate of $\frac{1}{i}$.

VI. Discussions

In this section, we discuss some practical issues.

Transmission credit mechanism: For each user k, our solution tells each node the optimized information rate $r_k(u)$=$\sum_{(u,v)\in E_k} r_k(u, v)$ and the optimized broadcast rate $b_k(u)$. We assume a node should be triggered to transmit only when it receives a packet, and should perform the transmission only when the MAC permits. We need a transmission credit $TX_k(u)$, computed as $TX_k(u)$=$\frac{b_k(u)}{r_k(u)}$ if $r_k(u)$>0 and 0 otherwise. Once u receives an innovative packet for user k from an upstream node, it increments the associated *credit counter* for user k by $TX_k(u)$. When a transmission is allowed by the MAC, u selects a user to serve by using a weighted round-robin algorithm [24] with each user k having a weight of $\frac{r_k(u)}{\sum_{i\in[1,K]} r_i(u)}$. u then checks whether the credit counter associated with this user is positive. If positive, it generates a coded packet, broadcasts it, and decrements the credit counter by one. Otherwise, it picks a different user. If all the credit counters are non-positive, u does not transmit. *Since a full-duplex MAC is still an open research area, our transmission credit mechanism also provides a possible way to integrate our work into some future full-duplex MACs.*

Computation for feasible minimum user rates: The minimum user rates should be carefully selected so that no user will require too large minimum user rate. Otherwise, the resource in the network may not satisfy them. We propose the following two computation methods. When the network is being deployed, the network administrator can use System 3 in Appendix B to initially analyze this network and compute the max-min user rate $\overline{\Lambda}$. It is easy to see that if each node sets its minimum user rate demand to a value less than $\overline{\Lambda}$, System 2 has a feasible solution. Although this method seems to use a central network administrator to solve System 3, it is still practical, since we only need to compute the max-min user rate *once* in the initial phase. Thus users still can adaptively adjust its minimum user rate demand when this network is being used. The nodes can also adjust their constraints as long as these constraints are not tighter than those used in System 3. Therefore, whenever the minimum user rate demands and node constraints change, the run-time optimization for System 2 can still be done in a distributed manner. Although the first method is very efficient, a central administrator is involved. An alternative method is using System 1 as a starting point to estimate the feasible minimum user rate demands. More specifically, users and nodes first run Pathbook-I to solve System 1. After enough iterations, they obtain a solution. Users can based on this solution estimate the feasible minimum user rate demands. In order to obtain fair minimum user rate demands, we can compute a Nash bargaining point [15] of user rates by setting $\mathcal{U}_k(R_k) = \ln R_k$ and $\boldsymbol{\lambda}_k(u) = 0$. The current solution to System 1 is a Nash bargaining point which captures the notion of social efficiency and fairness.

Information exchanges: We first consider the information exchanges for maintaining node load balance. To initially set up a load balance area, each node runs a distributed node discovery algorithm (such as [21]) to discover other nodes (such as k-hop neighbors). Through the analysis in Section V.B, we know that if the load balance area is too large, it will introduce many information exchanges. The following scheme can be used to reduce the information exchanges by roughly squeezing the overlap among the load balance areas of different nodes. Suppose that node u_1 wants to keep a balanced load with u_2 (i.e. $|b(u_1) - b(u_2)| \leq \boldsymbol{\theta}(u_1, u_2)$), and that u_1 knows another node u_3 (closer to u_1 than u_2 with respect to hops) has a constraint $|b(u_2) - b(u_3)| \leq \boldsymbol{\theta}(u_2, u_3)$ and $\boldsymbol{\theta}(u_1, u_2) \geq \boldsymbol{\theta}(u_2, u_3)$. Node u_1 hence sets $|b(u_1) - b(u_3)| \leq \boldsymbol{\theta}(u_1, u_3) = \min\{\boldsymbol{\theta}(u_1, u_3), \boldsymbol{\theta}(u_1, u_2) - \boldsymbol{\theta}(u_2, u_3)\}$, and excludes u_2 from its load balance area. The information exchange between u_1 and u_2 is hence not needed any more.

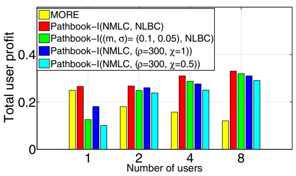
We now consider other information exchanges. In each iteration, each node broadcasts stepsize η, the iteration interval, the stop criteria, the information of the current iteration (i.e. computed user and node behaviors, and Lagrange multipliers). Other nodes can hence obtain the necessary information for the next iteration by overhearing others' broadcasts. Considering that the network topology and user sessions may change, we can divide the timeline into a series of intervals of a constant length. All the nodes and users roughly restart algorithms at the beginning of each interval.

Extensions: The analytical model and system used in this paper may be extensible when new issues and constraints are taken into account. Consider, for example, the following problem: user profit optimization problem subject to the user power consumption constraint $\sum_{u\in V_k\setminus d_k} \boldsymbol{c}(u)b_k(u) \leq \boldsymbol{c}_k$. We can add this constraint into System 1, and a similar distributed algorithm can be applied to solve this problem.

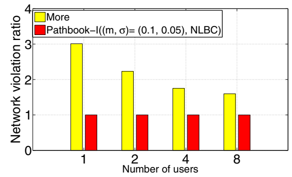
In this paper, we assume that all the nodes are equipped with full-duplex transceivers. If *some* nodes are equipped with traditional half-duplex transceivers, we only need to replace (8) with the traditional collision-free broadcast model (i.e., (4) in [44] or (3.3) in [14]). We can use similar distributed algorithms to solve these two optimization problems.

VII. Performance Evaluation

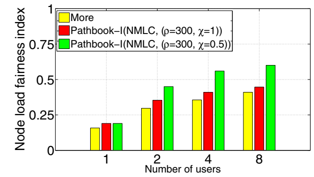
We implemented our OMR and compared its performance with that of an implementation of MORE [5]. Since to the best of our knowledge this work represents the first attempt towards the design of multipath routing optimization algorithms for full-duplex wireless networks, we assume that the nodes in MORE still adopt half-duplex transmission. We will implement MORE using full-duplex transceivers and evaluate its performance in the future work. We uniformly distributed 20 nodes in a $1000m\times1000m$ square region. As in [13], we assume that the packet delivery ratio from a node u to a node



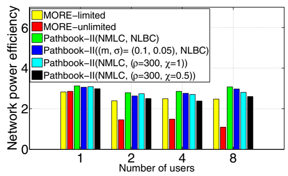
(a) Total user profit (System 1)



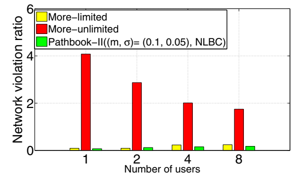
(b) Network violation ratio (System 1)



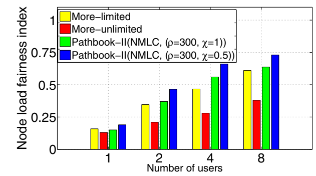
(c) Node load fairness index (System 1)



(d) Network power efficiency (System 2)



(e) Network violation ratio (System 2)



(f) Node load fairness index (System 2)

Fig. 2. Numerical results

v is inversely proportional to their distance $d(u,v)$ with a random Gaussian deviation of 0.1. If this packet delivery ratio is greater than 0.1, we say that v is in u's transmission range. We set the channel capacity $\mathbf{C}$ to 1Mbps. The numbers of users were chosen to be 1, 2, 4, and 8. For each user, we randomly chose two different nodes as its source and destination. In Fig.2, NMLC and NLBC denote the case where node max load constraints are not considered and the case where node load balance constraints are not considered, respectively. For node max load constraints, we normally generated each node u's $\boldsymbol{\varphi}(u)$ with a mean of mMbps and a variance of σMbps. Although u has its maximum load constraint $\boldsymbol{\varphi}(u)$, in our experiments we allow the real broadcast rates to be greater than $\boldsymbol{\varphi}(u)$. We compute a *violation ratio* $\frac{b(u)}{\boldsymbol{\varphi}(u)}$ for each node to quantify the constraint violation. We choose among all the nodes the worst (largest) violation ratio as the *network violation ratio*. In Fig.2, we use ρ to denote the load balance area range of each node. If $d(u,v)\leq\rho$, u(or v) is in v(or u)'s load balance area. In order to quantify the load balance improvement, we use the *fairness index* proposed in [23], defined by $\mathcal{F}=\frac{\left(\sum_{u\in V} b(u)\right)^2}{|V|\sum_{u\in V} b(u)^2}$.

Fig.2(a)-Fig.2(c) compare the results obtained by using MORE and Pathbook-I. We used $\ln(1+R_k)$ as user k's utility function, and the service price $\boldsymbol{\lambda}_k(u)$ was set to 0.01 per $Mbps$. If u (or v) was in v (or u)'s load balance area, $\boldsymbol{\theta}(u,v)$ and $\boldsymbol{\theta}(v,u)$ were set to $0.001\chi\times d(u,v)Mbps$, where the value of χ is shown in figures. In the experiments, all the users in MORE selfishly tried to transmit data as much as possible. Fig.2(a) shows that compared with MORE, Pathbook-I increases total user profit by 6%-175% when no node constraint is considered. Fig.2(b)-Fig.2(c) show that Pathbook-I achieves 37%-67% smaller network violation ratio, and 9%-57% higher node load fairness index. This significant performance improvement is expected, because our analytical system considers joint resource allocation by optimizing users' and nodes' behaviors.

Fig.2(d)-Fig.2(f) compare the results obtained by using MORE and Pathbook-II. The minimum user rate demands were set to 0.01Mbps for all the users. If u (or v) was in v (or u)'s load balance area, $\boldsymbol{\theta}(u,v)$ and $\boldsymbol{\theta}(v,u)$ were set to $0.0001\chi\times d(u,v)Mbps$, where the value of χ is shown in figures. The power consumption ratio $\boldsymbol{c}(u)$ was randomly chosen from (0, 1] for each node u. Note that although full-duplex transmission introduces more operations [9], we can still fairly assume that a node's power consumption ratios are the same in both half-duplex and full-duplex modes. This is because the major power consumption comes from the data transmission, and although two transmit antennas are being used by a full-duplex node [9], each transmit antenna radiates half the power in order to ensure the same total radiated power as with one transmit antenna used in the half-duplex operation. We define a metric, called *network power efficiency*, which is computed as $\sum_{k=1}^{K} R_k/\mathcal{N}(\overrightarrow{b})$. It represents the achieved total user rate per unit power consumption. In Fig.2(d)-Fig.2(f) "MORE-limited" and "MORE-unlimited" represent the case where all the users transmit data following their minimum user rate demands, and the case where all the users selfishly try to transmit data as much as possible, respectively. As shown in Fig.2(d)-Fig.2(f), we observe that compared with MORE-limited, Pathbook-II achieves 10%-24% higher network power efficiency (when no node constraint is considered), 21%-35% smaller network violation ratio, and 6%-41% higher node load fairness index (when the number of users is no less than 2). Compared with MORE-unlimited, Pathbook-II achieves 10%-183% higher network power efficiency, much smaller network violation ratio, and 15%-135% higher fairness index.

VIII. CONCLUSION

Due to the pioneer works by Choi *et al.* [9] and Jain *et al.* [22], building full duplex wireless networks (e.g. full-duplex 802.11n wireless networks) becomes possible. This paper has studied the cross-layer optimization for the opportunistic multipath routing in full-duplex wireless networks, comprehensively considering various resource competitions and constraints. We have proposed a collision-free full-duplex broadcast MAC and proved its necessary and sufficient conditions. We have concentrated on the user profit optimization problem and the network power consumption optimization problem, and have formulated these two problems as convex programming systems. By combining Lagrangian decomposition and subgradient methods, we have proposed distributed iterative algorithms to solve these two systems, which compute the optimized user information flow (i.e. user behavior) on the network layer and the optimized node broadcast rate (i.e. node behavior) on the MAC layer. Our algorithms allow each user and each node to adjust its behavior individually in each iteration. We have proved its convergence, and have also provided bounds on the amount of constraint violation, and the gap between our solution and the optimal solution in each iteration. Simulations results show that our algorithm can significantly improve user profit, reduce network violation ratio, increase node load fairness, and network power efficiency.

Our work provides a theoretical foundation for the future study on full-duplex wireless networks. To the best of our knowledge, this is the first work to study cross-layer optimization for full-duplex wireless networks.

REFERENCES

[1] D. Aguayo, J. Bicket, S. Biswas, G. Judd, and R. Morris; Link-level Measurements from an 802.11b Mesh Network; *ACM SIGCOMM'04*.

[2] Bay Area Wireless User Group. http://www.bawug.org.

[3] S. Biswas and R. Morris; ExOR: Opportunistic Multi-Hop Routing for Wireless Networks; *ACM SIGCOMM'05*.

[4] S. Boyd and L. Vandenberghe; Convex Optimization; *Cambridge University Press*.

[5] S. Chachulski, M. Jennings, S. Katti, and D. Katabi; Trading Structure for Randomness in Wireless Opportunistic Routing; *SIGCOMM'07*.

[6] J.-H. Chang and L. Tassiulas; Energy conserving routing in wireless ad-hoc nctworks." *IEEE INFOCOM'00*.

[7] J.-H. Chang and L. Tassiulas; Fast approximation algorithm for maximum lifetime routing in wireless ad-hoc networks; *in Lecture Notes in Computer Science; Networking 2000,* vol.1815 (2000).

[8] W. Cheng, X. Zhang, and H. Zhang; Full Duplex Wireless Communications for Cognitive Radio Networks; Available: http://arxiv.org/PS_cache/arxiv/pdf/1105/1105.0034v1.pdf.

[9] J.I. Choi, M. Jain, K. Srinivasan, P. Levis, S. Katti; Achieving Single Channel, Full Duplex Wireless Communication; *ACM MOBICOM'10*.

[10] D. D. Couto, D. Aguayo, J. Bicket, and R. Morris; A High-Throughput Path Metric for Multi-Hop Wireless Routing; *ACM MobiCom'03*.

[11] M. Duarte and A. Sabharwal; Full-duplex wireless communications using off-the-shelf radios: feasibility and first results; *Asilomar Conference on Signals, Systems, and Computers, 2010*

[12] E. Everett, M. Duarte, C. Dick, and A. Sabharwal; Empowering full-duplex wireless communication by exploiting directional diversity; *Submitted to Asilomar Conference on Signals, Systems, and Computers, 2011*.

[13] X. Fang, D. Yang, P. Gundecun and G. Xue; Multi-Constrained Anypath Routing in Wireless Mesh Networks; *IEEE SECON'10*.

[14] X. Fang, D. Yang, and G. Xue; Consort: Node-constrained Opportunistic Routing in Wireless Mesh Networks; *IEEE INFOCOM'11*.

[15] F. Forgo, J. Szep, and F. Szidarovszky; Introduction to the Theory of Games; *Kluwer Academic*, 1999.

[16] J. Gao and L. Zhang; Load balanced short path routing in wireless networks; *IEEE INFOCOM'04*

[17] Y. Ganjali and A. Keshavarzian; Load balancing in Ad hoc networks: single-path routing vs. multi-path routing; *IEEE INFOCOM'04*.

[18] S. Gollakota and D. Katabi; ZigZag decoding: combating hidden terminals in wireless networks; *ACM SIGCOMM'08*.

[19] S. Gollakota, H. Hassanieh, B. Ransford, D. Katabi, and K. Fu; They can hear your heartbeats: non-invasive security for implanted medical devices; *ACM SIGCOMM'11*.

[20] D. Halperin, T. Anderson, and D. Wetherall; Taking the sting out of carrier sense: interference cancellation for wireless lans; *MobiCom'08*.

[21] M. Harchol-Balter, T. Leighton, and D. Lewin; Resource Discovery in Distributed Networks; *ACM PODC'99*.

[22] M. Jain, J. I. Choi, T. M. Kim, D. Bharadia, S. Seth, K. Srinivasan, P. Levis, S. Katti, and P. Sinha; Practical, Real-time, Full Duplex Wireless; *ACM MOBICOM'11*.

[23] R. Jain, D. Chiu, and W. Hawe; A Quantitative Measure of Fairness and Discrimination For Resource Allocation in Shared Computer Systems; *DEC Research Technical Report*, 1984.

[24] M. Katevenis, S. Sidiropoulos, and C. Courcoubetis; Weighted Round-robin Cell Multiplexing in a General-purpose ATM Switch Chip; *IEEE JSAC*, 1991.

[25] F. Kelly; Charging and Rate Control for Elastic Traffic; *European Transactions on Telecommunications*, 1997.

[26] S. Kwon and N. B. Shroff; Paradox of Shortest Path Routing for Large Multi-hop Wireless Networks; *IEEE INFOCOM'07*.

[27] X. Li, Y. Wang, H. Chen, X. Chu, Y. Wu and Y. Qi; Reliable and Energy-Efficient Routing for Static Wireless Ad Hoc Networks with Unreliable Links; *IEEE Trans. on Parallel and Distributed Systems*'09.

[28] X. Lin, and N. B. Shroff; Utility Maximization for Communication Networks with Multipath Routing; *IEEE Trans. on automatic control*'06.

[29] M. Liu, J. Cao, G. Chen, and X. Wang; An Energy-Aware Routing Protocol in Wireless Sensor Networks; *Sensors* 2009.

[30] D. S. Lun, M. Médard, and R. Koetter; Network Coding for Efficient Wireless Unicast; *IEEE International Zurich Seminar on Comm.'06*.

[31] X. Mao, S. Tang, X. Xu, X. Li, and H. Ma: Energy Efficient Opportunistic Routing in Wireless Sensor Networks; *IEEE Transcations on parallel and distributed systems*'11.

[32] A. Mei, and J. Stefa; Routing in Outer Space: Fair Traffic Load in Multi-Hop Wireless Networks; *ACM MOBIHOC*'08.

[33] A. Nedić and A. Ozdaglar; Approximate Primal Solutions and Rate Analysis for Dual Subgradient Methods; MIT Technical Report, 2007.

[34] D. P. Palomar and M. Chiang; Alternative Distributed Algorithms for Network Utility Maximization: Framework and Applications; *IEEE Trans. on Automatic Control*, Vol. 52 (2007), pp.2254-2269.

[35] P. P. Pham and S. Perreau; Increasing the network performance using multi-path routing mechanism with load balance; *Ad Hoc Networks04*.

[36] L. Popa, A. Rostamizadeh, R. Karp, C. Papadimitriou, and I. Stoica; Balancing Traffic Load in Wireless Networks with Curveball Routing; *ACM MOBIHOC'07*.

[37] B. Radunović, D. Gunawardena, P. Key, A. Proutiere, N. Singh, H. V. Balan, and G. Dejean; Rethinking Indoor Wireless: Lower Power, Low Frequency, Full-duplex; *WiMesh (SECON Workshop)'10*.

[38] L. Rao, X. Liu, J. Chen, and W. Liu; Joint Optimization of System Lifetime and Network Performance for Real-Time Wireless Sensor Networks; *Quality of service in heterogeneous networks, Lecture notes of the institute for computer sciences, 2009*.

[39] A. Sahai, G. Patel, and A. Sabharwa; Pushing the limits of full-duplex: design and real-time implementation; *Technical Report TREE1104, Rice University*.

[40] A. Sankar and Z. Liu; Maximum Lifetime Routing in Wireless Ad-hoc Networks; *IEEE INFOCOM'04*.

[41] S. Singh. M. Woo. and C. S. Raghavzndra; Power-aware routing in mobile ad hoc network; *ACM/IEEE MOBICOM'98*

[42] A. Y. Wang and Charles G. Sodini; On the Energy Efficiency of Wireless Transceivers; *IEEE ICC 2006*.

[43] J. Wu, M. Lu, and F. Li; Utility-Based Opportunistic Routing in Multi-hop Wireless Networks; *ICDCS'08*

[44] X. Zhang and B. Li; Dice: a Game Theoretic Framework for Wireless Multipath Network Coding; *ACM MOBIHOC'08*.

[45] M. Zorzi and R. R. Rao; Geographic Random Forwarding (geraf) for Ad hoc and Sensor Networks: Energy and Latency Performance; *IEEE Transactions on Mobile Computing*, 2003.

APPENDIX A
PROOF OF THEOREM 5.1

Proof. First we prove the Slater's condition holds for System 1 and 2. For System 1, since $\boldsymbol{\varphi}(u) > 0, \boldsymbol{\theta}(u, v) > 0, \boldsymbol{p}(u, v) > 0$ and $\mathbf{C} > 0$, we can easily construct a *strictly* feasible solution by allocating each user a very small amount of information flow so that all the constraints are strictly satisfied. For System 2, we need to take care of the additional constraint: $R_k \geq \mathbf{\Lambda}_k, k \in [1, K]$. Section VI discusses how to compute a feasible minimum user rate. If we set $\mathbf{\Lambda}_k$ to a value which is strictly smaller than the corresponding minimum user rate computed in Section VI, the Slater condition also holds for System 2. Second, since the domain of each element in $\overrightarrow{x}$ is $[0, \mathbf{C}]$, we know $\overrightarrow{X}$ is a compact set. According to [33], we know that the norms of the subgradients of Systems 1 and 2 are bounded by a constant $\mathcal{L}$ (i.e. $\mathcal{L} = \max_{\overrightarrow{x} \in \overrightarrow{X}} ||g(\overrightarrow{x})||$). Since the Slater condition and the bounded subgradient condition hold, we can substitute the corresponding variables into Proposition 2 in [33] and finish the proof.

APPENDIX B
MAX-MIN USER RATE COMPUTATION

The following system provides a way to compute the max-min user rate. $System3(\overrightarrow{r}, \overrightarrow{b})$: maximize $\overline{\Lambda}$, subject to constraints (1)-(5) and $R_k \geq \overline{\Lambda}, \forall k \in [1, K]$.