Hindawi Publishing Corporation
EURASIP Journal on Wireless Communications and Networking
Volume 2009, Article ID 656832, 12 pages
doi:10.1155/2009/656832

Research Article

An Active Constraint Method for Distributed Routing, and Power Control in Wireless Networks

Alban Ferizi,[1] Armin Dekorsy,[2] Joerg Fliege,[3] Larissa Popova,[4] Wolfgang Koch,[4] and Michael Söllner[5]

[1] *Institute for Electronics Engineering, University Erlangen-Nuremberg, Cauerstraße 9, 91058 Erlangen, Germany*
[2] *Qualcomm CDMA Technologies GmbH, Nordostpark 89, 90411 Nuremberg, Germany*
[3] *CORMSIS, School of Mathematics, University of Southampton, University Road, Southampton, SO17 1BJ, UK*
[4] *Chair of Mobile Communications, University Erlangen-Nuremberg, Cauerstraße 7, 91058 Erlangen, Germany*
[5] *Alcatel-Lucent, Thurn-und-Taxis-Straße 10, 90411 Nuremberg, Germany*

Correspondence should be addressed to Alban Ferizi, ferizi@lfte.de

Received 17 April 2009; Revised 21 September 2009; Accepted 15 December 2009

Recommended by Alex B. Gershman

Efficiently transmitting data in wireless networks requires joint optimization of routing, scheduling, and power control. As opposed to the universal dual decomposition we present a method that solves this optimization problem by fully exploiting our knowledge of active constraints. The method still maintains main requirements such as optimality, distributed implementation, multiple path routing and per-hop error performance. To reduce the complexity of the whole problem, we separate scheduling from routing and power control, including it instead in the constraint set of the joint optimization problem. Apart from the mathematical framework we introduce a routing and power control decomposition algorithm that uses the active constraint method, and we give further details on its distributed application. For verification, we apply the distributed RPCD algorithm to examples of wireless mesh backhaul networks with fixed nodes. Impressive convergence results indicate that the distributed RPCD algorithm calculates the optimum solution in one decomposition step only.

1. Introduction

Nowadays, there is an increased interest in communication via wireless mesh networks such as ad-hoc, sensor, or wireless mesh backhauling networks [1, 2]. In wireless networks the link capacities are variable quantities and can be adjusted by the resource allocation such as scheduling, and power allocation to fully exploit network performance. Hence, for efficient data transmission an integrated routing, time scheduling and power control optimization strategy are required. This strategy has to take different transmission constraints into account, for example, maximum available power level or limited buffer size at nodes. The inherent decentralized nature of wireless mesh networks mandates that distributed algorithms should be developed to implement the joint routing, scheduling, and power control optimization. The first step towards a distributed implementation is to break up this problem into manageable subproblems and solve these subproblems by iterative algorithms. Cruz and Santhanam [3] have addressed the problem of finding an optimal link scheduling and power control policy while minimizing total average power consumption. Their algorithm is designed for single-path routing only, does not consider buffer limitations and has a worst case exponential complexity. In [4], Li and Ephremedis solve at first power control and scheduling jointly. They use the obtained power values to calculate a routing distance that in turn is used by Bellman-Ford routing. However, the proposed separation is performed by not considering the combinational structure of the entire routing, scheduling and power control problem. Although less computationally intensive, the algorithm ends up in a suboptimal solution. It further fully neglects multiple path routing as well as buffer restrictions. Xiao et al. proposed in [5] the dual decomposition as a promising decomposition approach. By dual decomposition the overall problem is split into two subproblems while the master dual problem coordinates them. In this paper we consider joint routing, time-scheduling, and power control for single frequency wireless mesh networks. The wireless transmissions are arranged in time-slots. However, we take into account that simultaneously active transmissions suffer from multiple access interference. Dual decomposition is a universal approach to solve such optimization problems [5, 6], but it does not consider the specific combinational structures of optimization problems. By contrast, we propose a novel method that explicitly exploits the combinational structure of a joint routing, time-scheduling, and power control problem by means of an active constraint method. The formulation of the optimization problem is yet generally valid, so that the method proposed here is applicable to a plurality of wireless networks. The proposed approach meets the following requirements: (1) less iterations to an optimum solution, (2) distributed implementation, (3) multiple path routing, and (4) per hop error performance. In particular, the approach is as follows. We separate scheduling from routing and power allocation by including it in the constraint set of a Simultaneous Routing and Power Control (SRPC) problem. For scheduling, several well known approximations such as Greedy-based approaches exist [7, Section 3.7], that we can leverage on. The constraints we use in the SRPC problem are induced by a precalculated colored graph of the network that, in turn, reflects the scheduling decisions of any arbitrary scheduler. Consequently, the main contribution is to introduce a Routing and Power Control Decomposition (RPCD) method to solve the simultaneous routing and power control problem while meeting the above-mentioned requirements. The clever bits of the RPCD are manifold. (1) We rewrite the SRPC problem to an equivalent problem by applying the active constraint method. (2) We decouple the equivalent problem by solving a (convex) network and a (convex) power assignment problem separately. (3) Iterations are performed by switching between the two subproblems for which network and power variables act as interchanging variables. Apart from the mathematical framework we introduce the RPCD algorithm and prove its convergence to a KKT-point of the joint routing and power control problem. We compare the RPCD algorithm with dual decomposition as state-of-art approach with respect to the number of iterations needed to calculate the KKT-point. This verification is performed by applying both algorithms to a wireless cellular mesh backhauling network [1, 2]. The backhauling network describes a "regular" cellular network. This models the situation where, in order to save infrastructure expenses of laying cable or fiber to each node (base station), we try to extend the range of a given source node with wired backhaul connection by using several other nodes. These intermediate nodes have no wired connection and can only communicate with the backhaul via the source node by wireless mesh communications. The simulation set-up correctly models mobile radio channel characteristics such as path-loss and slow fading. The comparison indicates that the RPCD approach requires only one decomposition step to calculate the optimum solution as opposed to dual decomposition. This paper is organized as follows. In Section 2 we describe the network model used for the wireless data network. In Section 3 we formulate the optimization problem and define the standard interference function. The RPCD algorithm for solving the joint routing and power control problem is presented in Section 4. We extend the RPCD algorithm in Section 5 by introducing distributed algorithms for solving the routing and power assignment problem. Finally, in Section 6 we apply the algorithm to a wireless backhaul network and present the simulation results. We conclude the paper in Section 7.

2. Network Model

The transmission problem we are facing is to transmit messages indexed by m each of size S_m in bits via a multiple hop wireless network. Each message has its source node s_m and its destination node $d_m \neq s_m$. Let M be an index set for the set of messages with $m \in M$. With *multiple path routing* each message is transmitted via several paths from its source node to its destination node. Thus, nodes can send parts of messages to many receivers and receive parts of messages from many transmitters. We denote the nodes by $v \in V$ with V be a finite set of nodes. At any time, each of these nodes $v \in V$ can map any parts of messages $m \in M$ onto a single link e for transmission. The set of all links is denoted by E. A wireless communication link corresponds to an edge $e = (u, v)$ between two nodes u, v and is described by the ordered pair $(u, v) \in V \times V$ such that u transmits information directly to v. Moreover, we assume that $(v, v) \notin E$ for all $v \in V$. We have that $G := (V, E)$ is a directed graph with node set V and edge set E. For an arbitrary node $v \in V$, denote by $E^{+}(v) := \{e \in E \mid e = (v, w) \in E\}$ and $E^{-}(v) := \{e \in E \mid e = (w, v) \in E\}$ the set of outgoing and incoming edges within E at the node v, respectively. A link represents a wireless resource characterized by a given bandwidth, time duration, space fraction, or by a given code assignment. We assume a time-slotted single frequency network for which the time is divided into equal slots of length τ while all nodes occupy the same frequency band of bandwidth B. Time slots are indexed by $t \in T$, with T as an index set. We take time scheduling into account by assuming that there is a given coloring of the nodes such that adjacent nodes do not have the same color (half-duplex constraint) [4]. That is, we are given a number C and a function $\mathrm{co}_V : V \to \{1, \ldots, C\}$ such that $\mathrm{co}_V(v) \neq \mathrm{co}_V(w)$ for all nodes $v, w \in V$ with $(v, w) \in E$. Here, C is at least as large as the chromatic number of G. Computing such a coloring can be done by a Greedy approach [7]. To take delay constraints into account we introduce $t_{\max}$ as the maximum number of time slots, a message is allowed to use for transmission from its source to its destination, that is, $T := \{1, \ldots, t_{\max}\}$. The interference model we consider includes multiple access interference caused by simultaneously active transmissions that can not be perfectly separated by, for example, code- or space division multiple access (CDMA/SDMA) techniques. Thus, let $E_{e,t}$ be the set of edges interfering edge e at time t. The signal attenuation from node u to node v is $G_t(u, v)$ and it remains unchanged within the duration of a time slot t. We further assume perfect knowledge of $G_t(u, v)$ at the corresponding senders. Let $T(e)$ be the transmitting node and let $R(e)$ be the receiving node of edge e. Hence, $G_t(T(l), R(e))$ denotes the attenuation a signal suffers that is transmitted from $T(l)$ but received by node $R(e)$. For link e such a signal represents multiple-access interference that is caused by link l. Furthermore, with $p_{e,t}$ as the (transmit) power to be allocated to link e at time slot t, the received signal power at node $R(e)$ from the transmitter $T(e)$ is given by $G_t(T(e), R(e))p_{e,t}$. We define the signal-to-interference-plus-noise ratio (SINR) of edge $e \in E$ at time slot $t \in T$ as

$$\mathrm{SINR}_{e,t} = \frac{G_t(T(e), R(e))p_{e,t}}{\sum_{l\in E_{e,t}, l\neq e} G_t(T(l), R(e))p_{l,t} + \sigma_e^2} \tag{1}$$

with σ_e^2 as an additive noise power of edge e. If we only assume thermal noise to be the same for all edges, we have $\sigma_e^2 = BN_0$ with noise spectral density N_0. For the optimization problems to be introduced later we have the following design variables. As network flow variables we have $c_{e,m,t} \in \mathbf{R}$ as the part of the message m sent along edge e in time slot t (in bits), and $b_{v,m,t} \in \mathbf{R}$ as the part of the message m stored in a buffer at node v directly before the start of time slot t (in bits). Communication variable $p_{e,t} \in \mathbf{R}$ is the transmit power allocated to edge e at time slot t to transmit the total traffic on edge e (in Watt). If we stack the different variables to vectors we obtain $\mathbf{c} = (c_{e,m,t})$, $\mathbf{b} = (b_{v,m,t})$, and $\mathbf{p} = (p_{e,t})$. We further use the following parameters. Let $S_m \in \mathbf{R}^+$ be the size of message m (in bits) and $B_v \in \mathbf{R}^+$ be the maximum total buffer size at node v (in bits). Power constraints are $P_v^{\max} \in \mathbf{R}^+$ as the maximum transmission power of a node (in Watt) assumed to be the same for all nodes and $P_e^{\max} \in \mathbf{R}^+$ as the maximum transmission power per edge (in Watt).

3. Optimization Problem

3.1. Problem Description. Let us consider an operation of a wireless data network with the objective to minimize a convex cost function $f(\mathbf{p}, \mathbf{c}, \mathbf{b})$ (or to maximize a concave utility function). The design variables **b**, **c,** and **p** are subject to some constraints. For instance, with $(e \in E, m \in M, v \in V, t \in T)$ we require the power constraints

$$p_{e,t} \geq 0, \tag{2}$$

$$p_{e,t} \leq P_e^{\max}, \tag{3}$$

$$\sum_{e\in E^+(v)} p_{e,t} \leq P_v^{\max} \tag{4}$$

forming the polyhedral set

$$C_p := \{\mathbf{p} \mid \mathbf{p} \text{ fulfills } (2),\ (3),\ \text{and } (4)\}. \tag{5}$$

Since we isolated coloring from the joint routing and power control, we have to take the precalculated colored network graph in the flow constraints into account. Similar to power constraints, we require that flow constraints form a polyhedral set C_c. For example, if we assume that given source nodes s_m have to transmit messages of sizes S_m to destinations d_m in a given time $t_{\max}$, the polyhedral set C_c is defined by the equalities and inequalities

$$c_{e,m,t} \geq 0 \quad (e \in E, m \in M, t \in T), \tag{6}$$

$$b_{v,m,t} \geq 0 \quad (v \in V, m \in M, t \in T), \tag{7}$$

$$b_{v,m,t} \leq B_{v,m} \quad (v \in V, m \in M, t \in T), \tag{8}$$

$$b_{s_m,m,1} = S_m \quad (m \in M), \tag{9}$$

$$b_{v,m,1} = 0 \quad (v \in V \setminus \{s_m\}, m \in M), \tag{10}$$

$$b_{d_m,m,t_{\max}} = S_m \quad (m \in M), \tag{11}$$

$$b_{v,m,t_{\max}} = 0 \quad (v \in V \setminus \{d_m\}, m \in M), \tag{12}$$

$$c_{e,m,t} = 0 \quad (m \in M, \mathrm{co}_E(e) \neq \mathrm{co}_T(t)), \tag{13}$$

$$\begin{gathered} b_{v,m,t+1} - b_{v,m,t} = \sum_{e\in E^-(v)} c_{e,m,t} - \sum_{e\in E^+(v)} c_{e,m,t}, \\ (m \in M, v \in V, t \in T \setminus \{t_{\max}\}). \end{gathered} \tag{14}$$

Equations (7) and (8) avoid buffer overload while (9) and (10) initialize buffer values. To account for delay constraints (11) and (12) ensure that messages reach their destinations completely at $t_{\max}$ at last . Coloring is ensured by (13) and (14) is a modified Kirchhoff's Law [6]. The SRPC problem under consideration is now as follows

$$\begin{aligned} \text{minimize} \quad & f(\mathbf{p}, \mathbf{c}, \mathbf{b}) \\ \text{subject to} \quad & (\mathbf{b}, \mathbf{c}) \in C_c \\ & \mathbf{p} \in C_p \\ & \sum_{m\in M} c_{e,m,t} \leq R_{e,t}(\mathbf{p}) \quad e \in E^+(v),\ t \in T. \end{aligned} \tag{15}$$

By the last constraints, we assume that at any time $t \in T$ each node $v \in V$ can map all part of messages $m \in M$ onto a single link $e \in E$ for transmission [5]. Furthermore, we assume that the amount of information (in bits) we can transmit on a single wireless link e at time slot t is bounded from above by a maximum mutual information bound $R_{e,t}(\mathbf{p})$ that itself depends on the power setting. The last constraints of (15) are the only constraints coupling network flow variables $(\mathbf{b}, \mathbf{c})$ with communication variables **p**. Thus, we call them coupling constraints [5], and they represent the most challenging constraints of the SRPC problem. All the other constraints are either constraints for the network flow variables or for the communication variables only. Assuming time-invariant channel conditions within the duration of a single time slot t, function $R_{e,t}(\mathbf{p})$ describes the amount of information of edge e and can be expressed with (1) by the well-known Shannon formula

$$R_{e,t}(\mathbf{p}) = B \cdot \tau \cdot \log_2\left(1 + \frac{1}{\Omega_e}\mathrm{SINR}_{e,t}\right) \quad (e \in E, t \in T). \tag{16}$$

For each edge e, the factor $\Omega_e \in \mathbf{R}_0^+$ represents any implementation margin relative to the maximum mutual information given by the Shannon formula [8]. In practice, achieving this mutual information requires adaptive modulation and coding. To accomplish the description of the optimization problem under consideration, we like to list some commonly used examples of cost (utility) functions. Examples are

(1) minimization of total transmitted power

$$f(\mathbf{p}) = \sum_{e\in E,\, t\in T} p_{e,t}, \tag{17}$$

(2) maximization of total network throughput

$$f(\mathbf{c}) = \sum_{v\in V} g_v(\mathbf{c}) \tag{18}$$

with $g_v(\mathbf{c})$ representing any linear combination of flows that egress node v,

(3) any linear combination of items (1) and (2).

A comprehensive overview on commonly used cost functions for wireless data networks is given in [5].

3.2. Standard Interference Function. In the following we give an interpretation of the coupling constraints ($\sum_{m\in M} c_{e,m,t} \leq R_{e,t}(\mathbf{p})$) of the SRPC problem (15). We define for $e \in E$ and $t \in T$

$$J_{e,t}(\mathbf{p},\mathbf{c}) := \frac{\Omega_e\left(2^{((\sum_{m\in M} c_{e,m,t})/B\cdot\tau)} - 1\right)}{G_t(T(e), R(e))} \times \left(\sum_{l\in E_{e,t},\, l\neq e} G_t(T(l), R(e))p_{l,t} + \sigma_e^2\right) \tag{19}$$

to be a *standard interference function* in $\mathbf{p}$ [9]. To clarify the meaning of a standard interference function, we restate the definition given in [9]. Here, $\succ$ and $\succeq$ mean componentwise inequality.

If we insert (1) into (16) and solve the coupling constraints in (15) for power values, we obtain rewritten coupling constraints as

$$\mathbf{p} \succeq \mathbf{J}(\mathbf{p},\mathbf{c}). \tag{20}$$

Definition 1. For given values $\hat{\mathbf{c}}$, $\mathbf{J}(\mathbf{p},\hat{\mathbf{c}})$ is a *standard interference function* if for all $\mathbf{p} \geq 0$ the following properties are satisfied.

(i) *Positivity*: $\mathbf{J}(\mathbf{p},\hat{\mathbf{c}}) \succ 0$.

(ii) *Monotonicity*: if $\mathbf{p} \succeq \mathbf{p}'$ then $\mathbf{J}(\mathbf{p},\hat{\mathbf{c}}) \succeq \mathbf{J}(\mathbf{p}',\hat{\mathbf{c}})$.

(iii) *Scalability*: For all $\alpha > 1$, $\alpha\mathbf{J}(\mathbf{p},\hat{\mathbf{c}}) \succ \mathbf{J}(\alpha\mathbf{p},\hat{\mathbf{c}})$

The positivity property ensures positive power values of the joint routing and power control problem (15). If any transmit power level is decreased, the monotonicity guarantees the decrease of the interference on the other links in the network, ensuring the maintenance of the same or even the achievement of a lower interference level for all links. The scalability property implies that if $\mathbf{p} \succeq \mathbf{J}(\mathbf{p},\hat{\mathbf{c}})$ then $\alpha\mathbf{p} \succeq \alpha\mathbf{J}(\mathbf{p},\hat{\mathbf{c}}) \succ \mathbf{J}(\alpha\mathbf{p},\hat{\mathbf{c}})$ for $\alpha > 1$.

Interestingly, (20) represents a Quality-of-Service (QoS) constraint, that is, a lower bound on the (implicitly defined) SINR. By reusing the coupling constraints again (15) and solving (16) for SINR we require

$$\mathrm{SINR}_{e,t} \geq \Omega_e\left(2^{((\sum_{m\in M} c_{e,m,t})/B\cdot\tau)} - 1\right) \quad (e \in E, m \in M, t \in T). \tag{21}$$

SINR is the main indicator for the transmission quality. Hence, given a modulation and coding scheme a specific per-hop error performance implies a respective Ω_e. In turn, by varying Ω_e we vary the transmission quality.

Note that for given values $\hat{c}_{e,m,t} \geq 0$, we can use a fixed point iteration algorithm to find a unique power vector $\mathbf{p}^* \in \mathbf{R}^{E\times T}$ with

$$\mathbf{p}^* = \mathbf{J}(\mathbf{p}^*,\hat{\mathbf{c}}). \tag{22}$$

This power iteration represents a standard power control algorithm as introduced in [9]. The power iteration used herein to solve (15) will be described in detail in Section 5.1. With coupling constraints (20) of problem (15) we can make use of the properties of the standard interference function [9], arriving at the following theorem.

Theorem 1. *Suppose that $\sigma_e^2 > 0$ ($e \in E$), that the objective function $f : (\mathbf{p},\mathbf{b},\mathbf{c}) \rightarrow f(\mathbf{p},\mathbf{b},\mathbf{c})$ is monotone in $\mathbf{p}$, and that we want to solve the optimization problem*

$$\begin{aligned} \textit{minimize} \quad & f(\mathbf{p},\mathbf{c},\mathbf{b}) \\ \textit{subject to} \quad & (\mathbf{b},\mathbf{c}) \in C_c \\ & \mathbf{p} \in C_p \\ & \mathbf{p} \succeq \mathbf{J}(\mathbf{p},\mathbf{c}). \end{aligned} \tag{23}$$

Suppose there exists a feasible point of this optimization problem, then there exists a feasible point with the same or better objective function value for which all the constraints (20) *are active, that is, equality holds in all of them. Especially, for every optimum objective function value there exists an optimum variable setting such that all constraints in* (20) *are active. If f is strictly monotone in* $\mathbf{p}$, *then all constraints* (20) *are active at each optimal solution of this problem.*

Proof. See Appendix A. □

Theorem 1 is an extension of the results found in [9]. In contrast to [9] we do not assume that f is just a sum of powers, instead it can be an arbitrary function being monotone in power values. Moreover, the objective as well as the coupling constraints depend on the flow variables $\mathbf{c}$ and buffers $\mathbf{b}$, a case not considered in [9].

4. RPCD-Algorithm

In this section we present the RPCD-Algorithm for solving the SRPC problem (15). In contrast to universal approaches,

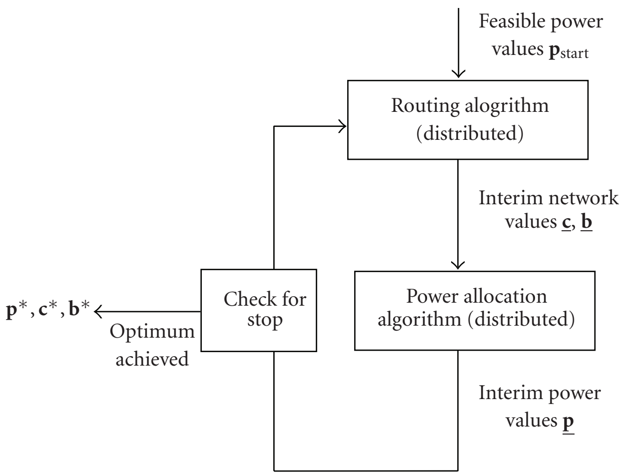

Figure 1: RPCD Algorithm.

like the dual decomposition method, we fully exploit our knowledge of active constraints of the joint optimization problem.

Based on Theorem 1 we can formulate an equivalent optimization problem but we avoid the extension of the utility function as usually done by applying dual or penalty approaches. We further keep the constraints and we only have to exchange the common network and power variables.

The main idea of the RPCD-Algorithm is to decouple the SRPC problem into two convex subproblems and to find the optimum solution of the SRPC problem by iteratively toggling between the two subproblems (Figure 1).

4.1. RPCD-Principle. Let us consider again problem (15). Due to Theorem 1 we know that all coupling constraints (20) of the SRPC problem are active at least at one optimum solution. By means of this observation, we can rewrite the SRPC problem to an equivalent problem as follows. Activity (equality) means that

$$\mathbf{p} = \mathbf{J}(\mathbf{p}, \mathbf{c}). \tag{24}$$

We now substitute (24) into the objective of the SRPC problem (15) and obtain an equivalent problem with the rewritten cost function as

$$\begin{aligned} \text{minimize} \quad & f(\mathbf{J}(\mathbf{p}, \mathbf{c}), \mathbf{c}, \mathbf{b}) \\ \text{subject to} \quad & (\mathbf{b}, \mathbf{c}) \in C_c, \\ & \mathbf{p} \in C_p, \\ & \mathbf{p} \succeq \mathbf{J}(\mathbf{p}, \mathbf{c}). \end{aligned} \tag{25}$$

In the following we use (25) and decompose the SRPC problem into two convex subproblems.

In particular, by assuming feasible power variables, a routing problem with fixed link capacities is formulated and the optimum flow variables for the routing problem are calculated. Equivalently, we can assume fixed routing variables and formulate a power control problem to calculate optimum power values [10].

The two subproblems are as follows.

4.1.1. Network Flow (Routing) Subproblem. We assume feasible power variables $\hat{\mathbf{p}} \in C_p$. With (25) we need to solve the optimization problem

$$\begin{aligned} \text{minimize} \quad & f(\mathbf{J}(\hat{\mathbf{p}}, \mathbf{c}), \mathbf{c}, \mathbf{b}) \\ \text{subject to} \quad & (\mathbf{b}, \mathbf{c}) \in C_c, \\ & \sum_{m \in M} c_{e,m,t} \le R_{e,t}(\hat{\mathbf{p}}), \end{aligned} \tag{26}$$

where $\mathbf{c}, \mathbf{b}$ are the optimization variables.

We have the following lemma.

Lemma 1. *(1) If f is a continuously differentiable and monotone function in $\mathbf{p}$ and in $\mathbf{c}$, then the objective of (26) is a continuously differentiable and monotone function in $\mathbf{c}$.*

(2) Let $\sigma_e^2 > 0$ $(e \in E)$. Suppose that f is twice continuously differentiable, that $f(\cdot, \mathbf{b}, \mathbf{c})$ is a convex and monotone function in $\mathbf{p}$ for all $(\mathbf{b}, \mathbf{c}) \in C_c$ and that $f(\mathbf{p}, \mathbf{b}, \cdot)$ is a convex function in $\mathbf{c}$ for all $\mathbf{p} \in C_p$, for all $\mathbf{b}$. Assume that at least one of the following holds:

(a) *$f(\cdot, \mathbf{b}, \mathbf{c})$ is strictly convex in $\mathbf{p}$ for all $(\mathbf{b}, \mathbf{c}) \in C_c$,*

(b) *$f(\mathbf{p}, \mathbf{b}, \cdot)$ is strictly convex in $\mathbf{c}$ for all $\mathbf{p} \in C_p$, for all $\mathbf{b}$,*

(c) *$f(\cdot, \mathbf{b}, \mathbf{c})$ is strictly monotone in $\mathbf{p}$ for all $(\mathbf{b}, \mathbf{c}) \in C_c$.*

Then, the objective of (26) is strictly convex in $\mathbf{c}$ and the solution to (26) is unique and continuous on $\hat{\mathbf{p}}$

Proof. See Appendix B □

4.1.2. Power Control Subproblem. We assume feasible network variables $\hat{\mathbf{c}}, \hat{\mathbf{b}} \in C_c$. We need to solve the optimization problem

$$\begin{aligned} \text{minimize} \quad & f\left(\mathbf{p}, \hat{\mathbf{c}}, \hat{\mathbf{b}}\right) \\ & \mathbf{p} \in C_p, \\ & \mathbf{p} \succeq \mathbf{J}(\mathbf{p}, \hat{\mathbf{c}}), \end{aligned} \tag{27}$$

where $\mathbf{p}$ are the optimization variables.

We have the following lemma.

Lemma 2. *Suppose that f is strictly monotone in $\mathbf{p}$ and (27) is feasible. Then, we have:*

(1) *problem (27) has a unique solution,*

(2) *the solution for (27) depends continuously on $(\hat{\mathbf{b}}, \hat{\mathbf{c}})$.*

Proof. See Appendix C □

4.2. RPCD Algorithm. As a consequence of the discussion above, we can replace the SRPC problem (15) by two simple subproblems, coupled to each other via fixed variables (power and network variables). The algorithmic scheme

Input: All parameters for problem (15).
Choose $\hat{\mathbf{p}}^{(0)} \in C_p$ sufficiently large so that problem (26) is feasible;
Choose $\hat{\mathbf{b}}^{(0)}$, $\hat{\mathbf{c}}^{(0)}$ arbitrarily;
$i := 0$;
while *stopping criterion for* $(\hat{\mathbf{b}}^{(i)}, \hat{\mathbf{c}}^{(i)}, \hat{\mathbf{p}}^{(i)})$ *not fulfilled* **do**
    Set $\hat{\mathbf{p}} := \hat{\mathbf{p}}^{(i)}$ and solve problem (26).
    Denote the result by $(\hat{\mathbf{b}}^{(i+1)}, \hat{\mathbf{c}}^{(i+1)})$.
    Set $(\hat{\mathbf{b}}, \hat{\mathbf{c}}) := (\hat{\mathbf{b}}^{(i+1)}, \hat{\mathbf{c}}^{(i+1)})$ and solve problem (27). Denote the result by $\hat{\mathbf{p}}^{(i+1)}$.
    $i := i + 1$.
endw
Output: $(\hat{\mathbf{b}}^{(i)}, \hat{\mathbf{c}}^{(i)}, \hat{\mathbf{p}}^{(i)})$

Algorithm 1: RPCD.

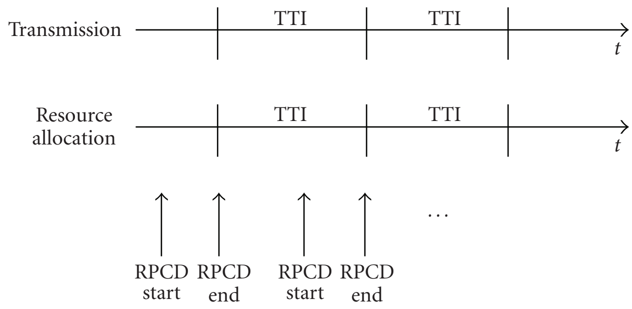

Figure 2: Time Alignment of RPCD with respect to transmission.

used is exemplified as RPCD algorithm and described in Algorithm 1.

Convergence of the RPCD algorithm is given by the following theorem.

Theorem 2. *Let us consider Lemmas 1 and 2. Under these assumptions and under the assumption that* (15) *is convex, the RPCD algorithm is well defined and provides a sequence of iterates* $(\hat{\mathbf{b}}^{(i)}, \hat{\mathbf{c}}^{(i)}, \hat{\mathbf{p}}^{(i)})_i$ *such that each subsequence of this sequence converges to an optimal point of* (15). *Moreover, there exists at least one converging subsequence. Additionally, the sequence* $(f(\hat{\mathbf{b}}^{(i)}, \hat{\mathbf{c}}^{(i)}, \hat{\mathbf{p}}^{(i)}))_i$ *converges monotonically decreasing.*

Proof. See Appendix D □

Note that both subproblems, (26) and (27), are *convex* and represent standard problems for which many efficient (distributed) algorithms exist. Particularly, we have to solve a flow problem with fixed capacities (fixed power values) [11] while computing optimum power values can be done by means of standard power control algorithms [9].

5. Distributed RPCD

Generally, we can apply centralized as well as distributed implementation for the RPCD algorithm. In this paper we concentrate on distributed algorithm exclusively. For the interested reader, a detailed survey about the centralized and distributed algorithms and their advantages and disadvantages can be found in [12].

Herein, for the distributed approach locally available information is required and we restrict the internode communication between neighbor nodes only.

As we illustrated in the Figure 2, each node executes the distributed RPCD algorithm in advance before a time slot begins. The algorithm allocates the resources optimally, for given network and power variables $\hat{\mathbf{c}}$ and $\hat{\mathbf{p}}$.

In the following, as introduced in Section 4.1, we consider again the two subproblems, routing (26) and power control (27), and present distributed algorithms for solving them.

5.1. Distributed Power Control. Let us consider again the power control subproblem (27), which is part of the RPCD algorithm.

$$\begin{aligned} \text{minimize} \quad & f\left(\mathbf{p}, \hat{c}, \hat{\mathbf{b}}\right) \\ & \mathbf{p} \in C_p, \\ & \mathbf{p} \succeq \mathbf{J}(\mathbf{p}, \hat{\mathbf{c}}). \end{aligned} \tag{28}$$

Assume given network variables $\hat{\mathbf{c}}, \hat{\mathbf{b}}$ and that the standard interference function $\mathbf{J}(\mathbf{p}, \hat{\mathbf{c}})$is feasible, that is, if the power vector $\mathbf{p} \in \mathbf{R}^+$ satisfies the coupling constraints (20), then we can use the following fixed point iteration to compute the optimum power settings

$$\mathbf{p}^{(n)} := \mathbf{J}\left(\mathbf{p}^{(n-1)}, \hat{c}\right) \quad n = 0, 1, 2, \ldots. \tag{29}$$

However, in practical systems we have $\mathbf{p} \in C_p$ taking power limitations into account. Given the original interference function $\mathbf{J}$ and considering the maximum power vector $P_e^{\max}$ in (3) we define

$$\mathbf{J}^{\mathbf{P}_e^{\max}}(\mathbf{p}, \hat{\mathbf{c}}) := \left(\min\{\mathbf{J}(\mathbf{p}, \hat{\mathbf{c}}), P_e^{\max}\}\right) \quad (e \in E^+(v), t \in T). \tag{30}$$

It has been proven in [9] that $\mathbf{J}^{\mathbf{P}_e^{\max}}$ is a standard interference function fulfilling Definition 1.

To include the constraint on the output power of a node, we define

$$G_{v,t} := \left\{ \left(p_{e,t}\right)_{e \in E^+(v)} \mid \sum_{e \in E^+(v)} p_{e,t} \le P_v^{\max} \right\} \tag{31}$$

and denote by $\operatorname{proj}_{G_{v,t}}^{e,t}$ a projection operator that maps computed power values into the polyhedral set C_p at each iteration step of the power iteration (29).

This projection allows us to consider only feasible power values during the course of the iteration.

By coupling the constrained interference function $\mathbf{J}^{\mathbf{P}_e^{\max}}$ with this projection on a polyhedral set, we define a new interference function $\mathbf{I}$ for given network variables $\hat{\mathbf{c}}$

$$I_{e,t}(\mathbf{p}, \hat{\mathbf{c}}) := \operatorname{proj}_{G_{v,t}}^{e,t}\left(\mathbf{J}^{\mathbf{P}_e^{\max}}(\mathbf{p}, \hat{\mathbf{c}})\right) \quad (e \in E^+(v), t \in T). \tag{32}$$

It can be easily shown that for all $\mathbf{p} \geq 0$ the interference function $\mathbf{I}(\mathbf{p}, \hat{\mathbf{c}})$ satisfies all properties given by Definition 1 and, hence, is also a standard interference function.

For each time step $t \in T$ we can now write the standard constrained power iteration as

$$\mathbf{p}^{(n)} := \mathbf{I}\left(\mathbf{p}^{(n-1)}, \hat{\mathbf{c}}\right) \quad n = 0, 1, 2, \ldots. \tag{33}$$

The power iteration (33) we call *distributed power control algorithm*.

Obviously, (33) is defined in terms of (32), (31), and (19). Due to (19), the information required to update the power values at starting node for a link $e \in E$ is the interference caused by the interfering transmissions measured at the end node for a link $e \in E$. Moreover, the projections introduced to consider the power constraints are local only. Hence, (33) represents a distributed power control algorithm [9].

We use (33) to find a unique vector $\mathbf{p}^* \in \mathbf{R}^{E \times T}$ with

$$\mathbf{p}^* = \mathbf{I}(\mathbf{p}^*, \hat{\mathbf{c}}). \tag{34}$$

If $\mathbf{I}(\mathbf{p}, \hat{\mathbf{c}})$ is feasible, then for any initial vector $\mathbf{p}$, the iteration (33) converges to a unique fixed point $\mathbf{p}^*$.

Due to Theorems 1 and 2, this unique fixed point of (34) is a solution of the SRPC problem (15). If the SRPC problem is (strictly) convex, the fixed point is the global (unique) solution for the power setting of the joint routing and power control problem.

5.2. Distributed Routing. In the following we present a distributed algorithm for solving the routing subproblem (26) with given power values $\hat{\mathbf{p}}$

$$\begin{aligned} \text{minimize} \quad & f(\mathbf{J}(\hat{\mathbf{p}}, \mathbf{c}), \mathbf{c}, \mathbf{b}) \\ \text{subject to} \quad & (\mathbf{b}, \mathbf{c}) \in C_c, \\ & \sum_{m \in M} c_{e,m,t} \leq R_{e,t}(\hat{\mathbf{p}}). \end{aligned} \tag{35}$$

The key to a distributed algorithm is to apply a decomposition method by means of formulating the dual problem of the optimization problem (26). Therefore we exploit the separable structure of the routing problem (26) via the dual decomposition method (see, e.g., [5, 13]). For solving the dual problem, we propose to apply the common approach of using the subgradient method [14].

To form the dual routing problem we rewrite the original routing problem (26) using the Lagrange function [6]. We introduce the Lagrange multipliers for the most involving constraints, which are the coupling constraints of the SRPC problem (15)

$$\sum_{m \in M} c_{e,m,t} \leq R_{e,t}(\hat{\mathbf{p}}) \tag{36}$$

and the flow conservation constraints, that is, modified Kirchhoff's Law (14)

$$\begin{gathered} b_{v,m,t+1} = b_{v,m,t} + \sum_{e \in E^-(v)} c_{e,m,t} - \sum_{e \in E^+(v)} c_{e,m,t}, \\ (m \in M, v \in V, t \in T \setminus \{t_{\max}\}). \end{gathered} \tag{37}$$

This results in the partial Lagrangian of (26) given as

$$\begin{aligned} L(\mathbf{c}, \mathbf{b}, \boldsymbol{\lambda}, \boldsymbol{\mu}) = & \sum_e \sum_t \frac{\Omega_e \left(2^{((\sum_{m \in M} c_{e,m,t})/B \cdot \tau)} - 1\right)}{G_t(T(e), R(e))} \\ & \cdot \left(\sum_{j \in E_{e,t} j \neq e} G_t(T(j), R(e)) p_{j,t} + \sigma_e^2 \right) \\ & + \sum_e \sum_t \mu_{e,t} \cdot \left(\sum_m c_{e,m,t} - R_{e,t}(\hat{\mathbf{p}}) \right) \\ & + \sum_v \sum_m \sum_t^{t_{\max}-1} \lambda_{v,m,t} \\ & \cdot \left(b_{v,m,t+1} - b_{v,m,t} - \sum_{e \in E^-(v)} c_{e,m,t} + \sum_{e \in E^+(v)} c_{e,m,t} \right) \end{aligned} \tag{38}$$

and the Lagrange multipliers are denoted by $\boldsymbol{\lambda} \in \mathbf{R}^{|V| \times |M| \times |T| - 1}$ and $\boldsymbol{\mu} \in \mathbf{R}^{E \times T}$.

The Lagrangian dual function is

$$\begin{aligned} V(\boldsymbol{\lambda}, \boldsymbol{\mu}) &= \inf_{\mathbf{c}, \mathbf{b}} L(\mathbf{c}, \mathbf{b}, \boldsymbol{\lambda}, \boldsymbol{\mu}) \\ \text{subject to} \quad & (b, c) \in C_c. \end{aligned} \tag{39}$$

Given the Lagrange dual function we can formulate the *dual problem* by [6]

$$\begin{aligned} D &= \sup_{\boldsymbol{\lambda}, \boldsymbol{\mu}} V(\boldsymbol{\lambda}, \boldsymbol{\mu}) \\ \text{subject to} \quad & \boldsymbol{\lambda} \text{ is arbitrary} \\ & \boldsymbol{\mu} \geq 0. \end{aligned} \tag{40}$$

We need to solve the dual problem (40) in order to obtain the best lower bound on $\mathbf{c}^*, \mathbf{b}^*$ from the Lagrange dual function (39). Since the Lagrangian dual function is convex, the dual problem is a convex optimization problem [5]. Moreover, Slater's condition (see, e.g., [6, 13]) holds and thus, strong duality holds. This means, the optimal value of the original routing problem (26) and the dual optimal value from (39) are equal and we can solve the primal problem (26) by its dual (40).

The algorithm to solve (39) and (40) is a two stage optimization algorithm. It solves (39) and (40) separately by using the subgradient method [6, 14] and toggling between the two subproblems until a convergence criterion is met.

For the computation of the dual function (39) we use the projected subgradient method [6, 14], which is an algorithm for minimizing a nondifferentiable convex function with the main feature of enabling distributed implementation.

As a first step we have to calculate the subgradients with the respect to the variables **c** and **b**, for variables $\boldsymbol{\lambda}$ and $\boldsymbol{\mu}$. These subgradients are given by

$$\mathrm{grad}^L_{b_{v,m,t}} = \frac{\partial L(\mathbf{c},\mathbf{b},\boldsymbol{\lambda},\boldsymbol{\mu})}{\partial b_{v,m,t}} = \lambda_{v,m,t-1} - \lambda_{v,m,t}, \tag{41}$$

$$\begin{aligned}\mathrm{grad}^L_{c_{e,m,t}} &= \frac{\partial L(\mathbf{c},\mathbf{b},\boldsymbol{\lambda},\boldsymbol{\mu})}{\partial c_{e,m,t}} \\ &= \frac{\Omega_e \cdot \left(\sum_{j\in E_{e,t},\, j\neq e} G_t(T(j),R(e))\, p_{j,t} + \sigma_e^2\right)\cdot \ln 2}{G_t(T(e),R(e))\cdot B\cdot\tau} \\ &\quad \cdot \left(2^{(\sum_{m\in M} c_{e,m,t})/B\cdot\tau}\right) + \mu_{e,t} + \lambda_{v^+(e),m,t} - \lambda_{v^-(e),m,t},\end{aligned} \tag{42}$$

where $v^+(e)$ denotes the node $v \in V$ that represents the starting point for one or more links $e \in E$. Analog to $v^+(e)$, we denote with $v^-(e)$ the node $v \in V$ that represents the end point for one or more links $e \in E$.

The subgradient updates on the variables $\mathbf{c}\,(e \in E, m \in M, t \in T)$ and $\mathbf{b}\,(v \in V, m \in M, t \in T)$ are

$$\begin{aligned} c^{(n+1)}_{e,m,t} &= \left[c^{(n)}_{e,m,t} - \alpha_n \mathrm{grad}^{L(n)}_{c_{e,m,t}}\right]^+, \\ b^{(n+1)}_{v,m,t} &= \left[b^{(n)}_{v,m,t} - \beta_n \mathrm{grad}^{L(n)}_{b_{v,m,t}}\right]^+. \end{aligned} \tag{43}$$

Note that the projection on the nonnegative orthant by $[\,]^+$ results due to the network constraints $c_{e,m,t} \geq 0$ $(e \in E, m \in M, t \in T)$ and $b_{v,m,t} \geq 0$ $(v \in V, m \in M, t \in T)$.

Furthermore, α_n and β_n represent the subgradient step sizes and have to satisfy (shown for α)

$$\lim_{n\to\infty} \alpha_n = 0, \qquad \sum_{n=1}^{\infty} \alpha_n = \infty \tag{44}$$

to ensure convergence. By $n = 1, 2, \ldots$ we denote the iteration step.

Finally, we have to solve the dual problem. For this, we compute the subgradients of the Lagrangian dual function $V(\lambda, \mu)$ due to the dual optimization variables $\lambda_{v,m,t}$ and $\mu_{e,t}$ that are given by

$$\begin{aligned}\mathrm{grad}^L_{\lambda_{v,m,t}} &= \frac{\partial}{\partial \lambda_{v,m,t}} \inf_{\mathbf{c},\mathbf{b}} L(\mathbf{c},\mathbf{b},\boldsymbol{\lambda},\boldsymbol{\mu}) \\ &= b_{v,m,t+1} - b_{v,m,t} \\ &\quad - \sum_{e\in E^-(v)} c_{e,m,t} + \sum_{e\in E^+(v)} c_{e,m,t}, \\ \mathrm{grad}^L_{\mu_{e,t}} &= \frac{\partial}{\partial \mu_{e,t}} \inf_{\mathbf{c},\mathbf{b}} L(\mathbf{c},\mathbf{b},\boldsymbol{\lambda},\boldsymbol{\mu}) \\ &= \sum_m c_{e,m,t} - R_{e,t}(\hat{\mathbf{p}}).\end{aligned} \tag{45}$$

Applying subgradient update we obtain for variables $\boldsymbol{\lambda}(v \in V, m \in M, t \in T)$ and $\boldsymbol{\mu}(e \in E, t \in T)$

$$\begin{aligned}\lambda^{(n+1)}_{v,m,t} &= \left(\lambda^{(n)}_{v,m,t} + \delta_n \mathrm{grad}^{L(n)}_{\lambda_{v,m,t}}\right), \\ \mu^{(n+1)}_{e,t} &= \left[\mu^{(n)}_{v,m,t} + \epsilon_n \mathrm{grad}^{L(n)}_{\mu_{e,t}}\right]^+.\end{aligned} \tag{46}$$

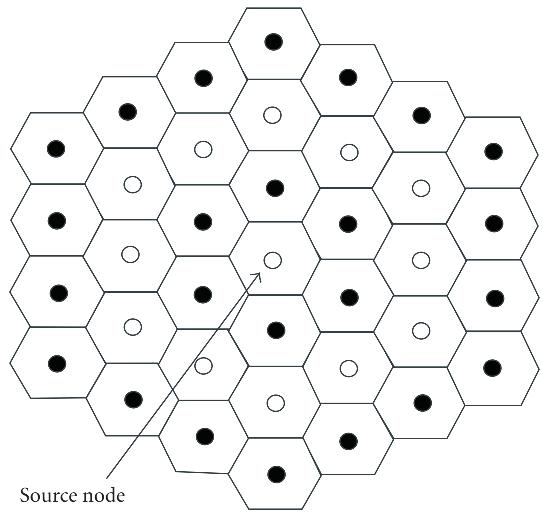

Figure 3: Wireless Mesh Backhaul Network.

Note that the projection on the nonnegative orthant by $[\,]^+$ results due to the constraints $\mu_{e,t} \geq 0$ $(e \in E, t \in T)$.

Furthermore, δ_n and ϵ_n represent the subgradient step sizes, both satisfying the conditions in (44) with $n = 1, 2, \ldots$ denoting the iteration step.

As one can see by considering (33), (41), (42), (45), (39), (46), and (41) two types of information are necessary. First, that the information required for the computation to take place at each and every node is the interference caused by the interfering transmissions measured at the receiving node. Second, by (42) Lagrange multipliers from neighbor nodes, for example, $v^-(e)$ and $v^+(e)$ are required.

The distributed routing algorithm tries to achieve an optimum coordination between the network variables **c** and **b** on the one hand and the dual variables $\boldsymbol{\lambda}$ and $\boldsymbol{\mu}$ on the other hand. For the considered wireless network, this means that the distributed routing algorithm tries to achieve an optimum coordination between node buffers and capacities allocated to the links, subject to the network constraints as defined in (26).

6. Simulation Results

In this section, we present some numerical results of the distributed RPCD algorithm as applied to a wireless mesh backhaul network. Furthermore, we compare the results with the dual decomposition method introduced by Xiao et al. in [5]. The network under consideration is a typical cellular network with hexagonal cell structure. The cells are arranged around a center cell by rings and a node is located in the center of a hexagon as depicted in Figure 3.

This models the situation where, to save infrastructure expenses like laying cable or fiber to each node in a network, we try to extend the range of given source node (center node) by intermediate nodes being wireless connected. The source node has wired backhaul connection only, while all other nodes have no wired backhaul connection and can only communicate with the wireless mesh backhaul via the source

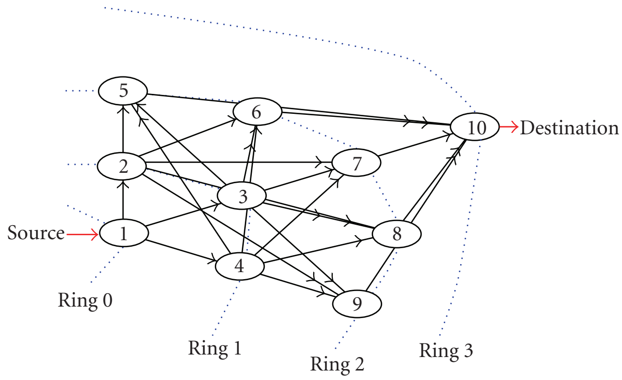

FIGURE 4: Network [1 3 5 1].

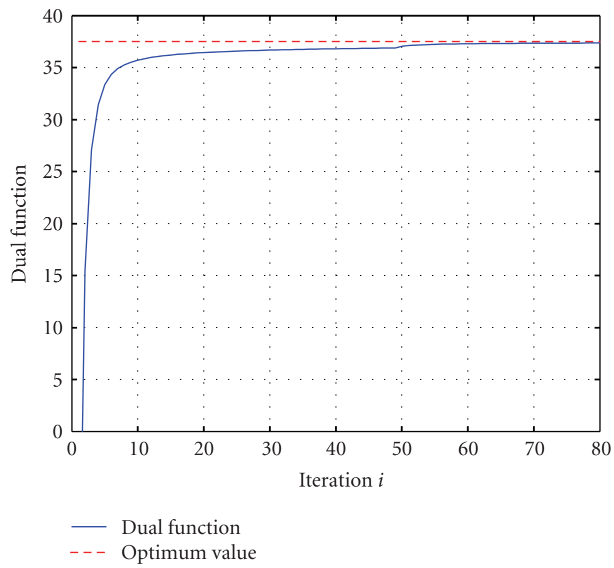

FIGURE 5: Dual Decomposition: Progress of dual function for [1 3 5 1] network.

node. We require that wireless links can only be formed between nodes in adjacent rings. This means, (1) a node can not transmit to any node that is more than one ring away, and (2) intraring communication is not allowed so that nodes belonging to the same ring have no wireless link established. Figure 4 shows the resulting directed graph of the wireless mesh backhaul network for the case where the first ring composes three, the second ring five, and the third ring the destination node only.

Each intermediate node can transmit to and receive along multiple links from nodes, neither multicast nor broadcast is considered. The network is a single frequency network. For the sake of simplicity, we assume that the scheduler does not take in-band signaling users into account, rather we might interpret in-band users as additive noise. We further require in the simulations that simultaneously active links do not interfere. Hence, the SRPC problem under consideration is convex, therefore, the optimum solution is global. This means, we assume orthogonal transmission between links, possibly performed by Space Division Multiple Access (SDMA) schemes such as sending/receiving beamforming [15, 16]. Due to the setup of the wireless backhaul links, the nodes we consider are cellular base stations with high processing capability. Without loss of generality, the objective function we assume is to minimize total transmitted power with $f(\mathbf{p}) = \sum_{e\in E, t\in T} p_{e,t}$.

The scenario shown in Figure 4 is denoted as [1, 3, 5, 1] scenario. So, we have 10 nodes forming a wireless mesh backhauling network with 23 edges. The simulation parameter set up is as follows. The wireless network has to transmit data of $S_m = 10$ Mbit size from the source node to the destination, but due to the delay constraint the transmission has to be completed within a maximum number of $t_{\max} = 7$ time slots, that is, $T = \{1, \ldots, 7\}$. The bandwidth per link is $B = 5$ MHz, the length of an time-slot is $\tau = 1$ ms and the radius per hexagonal cell is $r = 500$ m. We assume an exponential path-loss model with factor 3, but no shadow-fading. The thermal spectral noise density is $\sigma^2 = -174$ dBm/Hz. The buffer size per node is restricted to $B_{v,m} = 10$ Mbit. To account for power constraints we upper bound the power per node by $P_v^{\max} = 10$ Watt, whereas for each specific link we assume no explicit power restriction. Regarding the distributed RPCD algorithm, we use a stopping criterion based on the variables $c_{e,m,t} (e \in E, m \in M, t \in T)$ to check for convergence. We proceed with the iteration as long as the maximum norm of two consecutive iteration steps is greater than 10^{-7}. To show convergence, we apply the algorithm for a huge number of different starting points as well as for several networks, that is, [1, 3, 1], [1, 3, 5, 1], [1, 3, 5, 7, 1].

We observe the following result.

For a given feasible starting point $\hat{\mathbf{p}}^{(0)}$ *the distributed RPCD algorithm converges to an optimum solution within one step only.*

Since the algorithm converges globally the zero vector is always a feasible starting point.

The optimum solution is cross checked twice. First, we verify the solution by applying the NPSOL solver of TOMLAB that reflects centralized implementation. Secondly, we compare our results with another distributed algorithm, the dual decomposition approach [5]. Figure 5 shows the dual function versus iteration i. Clearly, the dual function slowly converges to the unique optimum solution (as proposed in [5], we applied the subgradient method to update the dual variables).

Hence, it is obvious that the proposed method significantly outperforms the dual decomposition approach in terms of required iterations steps towards the optimum solution. Moreover, the distributed RPCD approach requires an inter-node communication where nodes share the power and network variables only. The dual decomposition, however, requires an extra communication of the dual variables [17]. Finally, Figure 6 shows the average rate allocation of the data values per edge, where averaging is performed over the time slots the links are active. The amount of transmitted bits is given in [Mbit] while the power values are given in [Watt]. For illustration purposes, the thickness of the links

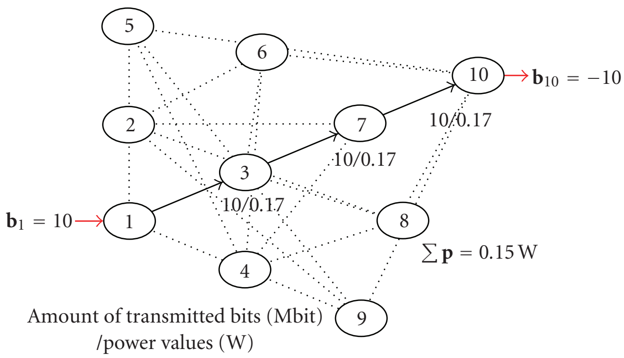

FIGURE 6: Network [1, 3, 5, 1]: Average rate allocation, Bandwidth $B = 5$ MHz.

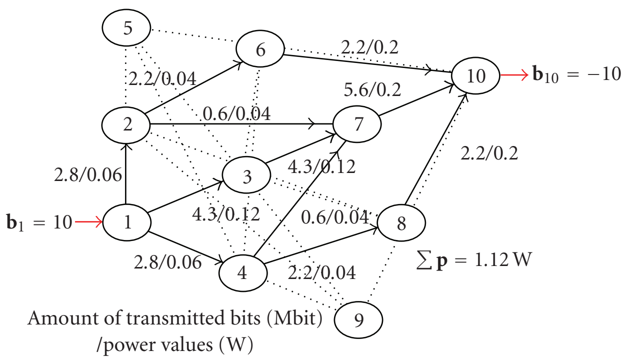

FIGURE 7: Network [1, 3, 5, 1]: Average rate allocation, Bandwidth $B = 1$ MHz.

reflects the amount of data transmitted, while dotted links are never active during the entire transmission. As expected, we observe that due to the geometry of the network traffic is mainly concentrated in inner links and the algorithm use one single route from the source to the destination, although multiple path routing could be performed.

Further, we decrease the bandwidth for every link in the network and use $B = 1$ MHz. In Figure 7 we can observe that the algorithm can not transmit the total amount of the data over one single route anymore and *multiple path routing* has to be performed. The transmission of the data expands over more routes in the wireless mesh backhaul network. Nevertheless the data traffic is generally concentrated in the inner links, which can be explained with the geometry of the network.

7. Conclusion

In this paper we have considered the joint routing, time scheduling and power control problem for single frequency, time-slotted wireless mesh networks. We presented an approach for optimally solving this crosslayer optimization problem while meeting the requirements, such as distributed implementation, multiple path routing, and per-hop error performance. The main contribution is the distributed Routing and Power Control Decomposition (RPCD) Algorithm, which is based on the idea of decoupling the SRPC problem into two subproblems, power control and routing, and including scheduling in the constraint set of the SRPC problem. Moreover, we presented distributed algorithms for solving both, the power control and the routing subproblem. For illustration purpose we applied the distributed RPCD algorithm to a wireless mesh backhaul network. The observed convergence results are impressive: only *one* decomposition step is needed to achieve the optimal solution.

Appendices

A. Proof of Theorem 1

Proof. Choosing feasible variables $p_{e,m,t}, c_{e,m,t}, b_{v,m,t}$ ($e \in E, m \in M, t \in T, v \in V$) for the problem above, we immediately see that the interference function defined by (29) is a standard interference function. Therefore, the power iteration (29) started from, say, $\mathbf{p}^{(0)} = \mathbf{0}$ converges to a point $\mathbf{p}^*$ with (22). Clearly, for this point the constraints (20) are active. Moreover, in [9] it was shown that $\mathbf{p}^*$ has the smallest objective function value for all possible choices of the variables $p_{e,m,t}$ ($e \in E, m \in M, t \in T, v \in V$) with prespecified and fixed variables $c_{e,m,t}, b_{v,m,t}$ ($e \in E, m \in M, t \in T, v \in V$). Therefore, the constraints (20) have to be active in all solutions of the optimization problem above. □

B. Proof of Lemma 1

Proof. (1) This can be seen by differentiating the objective under consideration with respect to $\mathbf{c}$.

(2) The Hessian of the objective under consideration with respect to $\mathbf{c}$ can be written as

$$\left(\frac{\partial \mathbf{J}}{\partial \mathbf{c}}\right)^T \frac{\partial^2 f}{\partial \mathbf{p}^2} \frac{\partial \mathbf{J}}{\partial \mathbf{c}} + \mathbf{D} + \frac{\partial^2 f}{\partial \mathbf{c}^2}, \tag{B.1}$$

where $\mathbf{D}$ is a diagonal matrix with entries

$$\frac{\partial f}{\partial p_{e,m,t}} \frac{\partial^2 J_{e,m,t}}{\partial^2 c_{e,m,t}}. \tag{B.2}$$

The objective is strict convex in $\mathbf{c}$ if and only if this Hessian is positive definite. Now, the first summand above is clearly positive semidefinite, since $f(\cdot, \mathbf{b}, \mathbf{c})$ is convex in $\mathbf{p}$. Likewise, the third summand is positive semidefinite, since $f(\mathbf{p}, \mathbf{b}, \cdot)$ is convex in $\mathbf{c}$. Finally, the diagonal matrix $\mathbf{D}$ has nonnegative entries on the main diagonal, since $\mathbf{J}$ is strictly convex in $\mathbf{c}$ and $f(\cdot, \mathbf{b}, \mathbf{c})$ is monotone in $\mathbf{p}$. Accordingly, the Hessian above is positive definite as long as one of the summands is positive definite. Assumption (a) leads to the positive definiteness of the first summand, assumption (b) leads to the positive definiteness of the last summand, while assumption (c) leads to the positive definiteness of $\mathbf{D}$.

From this, the strict convexity under the given assumptions readily follows.

Clearly, the variables $c_{e,m,t}$ ($e \in E$, $m \in M$, $t \in T$) are unique in an optimum of the problem under consideration. Using (9), (10), and (14) and induction over $t \in T$, one easily concludes that optimal variables $b_{v,m,t}$ ($v \in V, m \in M, t \in T$) are unique, too.

The continuity is a classic result from parametric optimization, see, for example, [18], and follows from the strict convexity of the objective in $\mathbf{c}$ and from the fact that the set of feasible points is a polyhedron. □

Remark 1. The following remarks hold with respect to Lemma 1.

(1) The second result of Lemma 1 can be weakened a bit. In case f is not strictly convex in, say, $\mathbf{c}$, it suffices to assume that the diagonal entries of the matrix $\mathbf{D}$ are sufficiently large. That amounts to saying that either $c_{e,m,t}$ is sufficiently large or that $\partial f / \partial p_{e,m,t}$ is sufficiently large, that is, there is certain amount of monotonicity build into the objective function ($e \in E, m \in M, t \in T$).

(2) The convexity assumptions of Lemma 1 do *not* amount in assuming that f is convex, as the example $f(p,c) = pc$ shows.

C. Proof of Lemma 2

Proof. (1) See Theorem 1 and [9].

(2) This follows by noting that the solution to the problem at hand is uniquely characterized by the fixed-point equation $\mathbf{p} = \mathbf{J}(\mathbf{p}, \hat{\mathbf{c}})$, the latter being a linear system in $\mathbf{p}$. The corresponding solution depends continuously on $(\hat{\mathbf{b}}, \hat{\mathbf{c}})$. □

Remark 2. In Lemma 2, strict monotonicity cannot be replaced by monotonicity. However, uniqueness of the results of Lemma 2 hold again if only the (unique) $\|\cdot\|_2$-solution of the optimization problem under consideration is considered.

D. Proof of Theorem 2

Proof. The well-definedness rests on the two lemmas above, which also tell us that the maps $\mathbf{p} \mapsto \mathrm{P}(f, \mathbf{p})$ and $(\mathbf{b}, \mathbf{c}) \mapsto \mathrm{P}(f, \mathbf{b}, \mathbf{c})$, mapping parameters to solutions of optimization problems, that are considered in the algorithm are point-to-point. Moreover, both maps are continuous. We call an arbitrary mapping $\mathcal{M} : \mathbf{x} \mapsto \mathcal{M}(\mathbf{x})$*closed* if $\lim_{k\to\infty} \mathbf{x}^{(k)} = \mathbf{x}$ and $\lim_{k\to\infty} \mathcal{M}(\mathbf{x}^{(k)}) = y$ imply $\mathcal{M}(\mathbf{x}) = y$. Clearly, continuous mappings are closed, and therefore the two mappings mentioned above are closed. (See also [19, page 123],. Actually, closedness is a property usually associated with point-to-set mappings, but we only consider point-to-point mappings here. For point-to-point mappings, continuity is sufficient for closedness, and these two notions are equivalent if the set of arguments is compact.) The rest of the theorem follows the convergence proof of the coordinate descent method [19, Section 7.8], replacing the set $\{\mathbf{x} \mid f(\mathbf{x}) = \mathbf{0}\}$ with the set of feasible points for which there does not exist a feasible direction of descent: since C_p is compact, the coupling constraint implies that C_c is compact, too, and therefore the whole set of feasible points is compact. Accordingly, the composition of maps

$$\begin{aligned}\mathcal{M} : \left(\mathbf{p}^{(0)}, \mathbf{b}^{(0)}, \mathbf{c}^{(0)}\right) &\mapsto \left(\mathbf{p}^{(0)}, P\left(f, \mathbf{p}^{(0)}\right)\right) =: \left(\mathbf{p}^{(1)}, \mathbf{b}^{(1)}, \mathbf{c}^{(1)}\right) \\ &\mapsto \left(P\left(f, \mathbf{b}^{(1)}, \mathbf{c}^{(1)}\right), \mathbf{b}^{(1)}, \mathbf{c}^{(1)}\right) \\ &=: \left(\mathbf{p}^{(2)}, \mathbf{b}^{(2)}, \mathbf{c}^{(2)}\right).\end{aligned} \tag{D.1}$$

that is,

$$\mathcal{M}\ (\mathbf{p}, \mathbf{b}, \mathbf{c}) := (P(f, P(f, \mathbf{p})), P(f, \mathbf{p})) \tag{D.2}$$

is closed, see [19, page 124]. That the sequence $(f(\hat{\mathbf{b}}^{(i)}, \hat{\mathbf{c}}^{(i)}, \hat{\mathbf{p}}^{(i)}))_i$ is monotonically decreasing (and therefore convergent) follows by construction of the sequence. We can now define the compact set

$$\begin{aligned}\Gamma &:= \{(\mathbf{p}, \mathbf{b}, \mathbf{c}) \mid (\mathbf{p}, \mathbf{b}, \mathbf{c}) \text{ feas. } \& \ f(\mathbf{p}, P(f, \mathbf{p})) \\ &\qquad \geq f(\mathbf{p}, \mathbf{b}, \mathbf{c}) \ \& \ f(P(f, \mathbf{b}, \mathbf{c}), \mathbf{b}, \mathbf{c}) \geq f(\mathbf{p}, \mathbf{b}, \mathbf{c})\} \\ &= \{(\mathbf{p}, \mathbf{b}, \mathbf{c}) \mid (\mathbf{p}, \mathbf{b}, \mathbf{c}) \text{ feas. } \& \ f(\mathbf{p}, P(f, \mathbf{p})) \\ &\qquad = f(\mathbf{p}, \mathbf{b}, \mathbf{c}) \ \& \ f(P(f, \mathbf{b}, \mathbf{c}), \mathbf{b}, \mathbf{c}) = f(\mathbf{p}, \mathbf{b}, \mathbf{c})\},\end{aligned} \tag{D.3}$$

that is, the set of feasible points from which the algorithm does not improve objective function values any more. In the convex case this is a set of KKT-points which is the same as the set of the optimal points. It is easy to see that every converging subsequence of $(\hat{\mathbf{p}}^{(i)}, \hat{\mathbf{b}}^{(i)}, \hat{\mathbf{c}}^{(i)})_i$ converges to a point in Γ, more precisely to a fixed point of $\mathcal{M}$ (see also [20]), which is thereby a point for which no feasible direction of descent exists. □

Remark 3. The following remarks hold with respect to Theorem 2.

(1) The convexity assumptions within the theorem have mainly be imposed to guarantee that the map $\mathbf{p} \mapsto \mathrm{P}(f, \mathbf{p})$ is well-defined (i.e., the solution to the corresponding optimization problem is unique). If we simply assume this well-definedness (or enforce it by, say, computing the least-squares optimal solution), we can drop the corresponding assumptions on f instead.

(2) In principle, strict monotonicity of f in $\mathbf{p}$ is required to obtain that $(\mathbf{b}, \mathbf{c}) \mapsto \mathrm{P}(f, \mathbf{b}, \mathbf{c})$ is continuous. However, what usually happens is that in step 5 the power iteration is used, which results in the computation of the unique $\|\cdot\|_1$-solution to $\mathrm{P}(f, \mathbf{b}, \mathbf{c})$. In case the power iteration is replaced by another algorithm, uniqueness of the corresponding solution has to be guaranteed in a different way.

References

[1] H. Viswanathan and S. Mukherjee, "Throughput-range tradeoff of wireless mesh backhaul networks," *IEEE Journal on Selected Areas in Communications*, vol. 24, no. 3, pp. 593–602, 2006.

[2] "The IEEE 802.16 Working Group on Broadband Wireless Access Standards," http://www.wirelessman.org/.

[3] R. L. Cruz and A. V. Santhanam, "Optimal routing, link scheduling and power control in multi-hop wireless networks," in *Proceedings of the 22nd Annual Joint Conference on the IEEE Computer and Communications Societies (INFOCOM '03)*, vol. 1, pp. 702–711, San Francisco, Calif, USA, March-April 2003.

[4] Y. Li and A. Ephremides, "Joint scheduling, power control, and routing algorithm for ad-hoc wireless networks," in *Proceedings of the 38th Annual Hawaii International Conference on System Sciences*, p. 322, Big Island, Hawaii, USA, January 2005.

[5] L. Xiao, M. Johansson, and S. P. Boyd, "Simultaneous routing and resource allocation via dual decomposition," *IEEE Transactions on Communications*, vol. 52, no. 7, pp. 1136–1144, 2004.

[6] S. Boyd and L. Vandenberghe, *Convex Optimization*, Cambridge University Press, Cambridge, UK, 2004.

[7] S. B. Maurer and A. Ralston, *Discrete Algorithmic Mathematics*, Addison-Wesley, Reading, Mass, USA, 1991.

[8] M. V. Eyuboglu and G. D. Forney Jr., "Trellis precoding: combined coding, precoding and shaping for intersymbol interference channels," *IEEE Transactions on Information Theory*, vol. 38, no. 2, part I, pp. 301–314, 1992.

[9] R. D. Yates, "Framework for uplink power control in cellular radio systems," *IEEE Journal on Selected Areas in Communications*, vol. 13, no. 7, pp. 1341–1347, 1995.

[10] A. Dekorsy, J. Fliege, and M. Sollner, "Optimal distributed routing and power control decomposition for wireless networks," in *Proceedings of IEEE Global Telecommunications Conference (GLOBECOM '07)*, pp. 4920–4924, Washington, DC, USA, November 2007.

[11] D. P. Bertsekas and R. G. Gallager, *Data Networks*, Prentice-Hall, Englewood Cliffs, NJ, USA, 1992.

[12] D. P. Bertsekas, *Parallel and Distributed Computation*, Prentice-Hall, Englewood Cliffs, NJ, USA, 1989.

[13] D. P. Bertsekas, *Nonlinear Programming*, Athena Scientific, Nashua, NH, USA, 1999.

[14] A. Ferizi, *Distributed routing and power control algorithms in fixed wireless networks*, Diploma Thesis, University Erlangen-Nuremberg, Erlangen, Germany, 2007.

[15] A. Paulraj, R. Nabar, and D. Gore, *Introduction to Space-Time Wireless Communications*, Cambridge University Press, Cambridge, UK, 2003.

[16] A. Czylwik, A. Dekorsy, and B. Chalise, "Smart antenna solutions for UMTS," in *Smart Antennas–State of the Art*, vol. 3 of *EURASIP Book Series on Signal Processing and Communications*, chapter 4, pp. 729–758, Hindawi Publishing Corporation, Cairo, Egypt, 2005.

[17] M. Chiang, "Balancing transport and physical layers in wireless multihop networks: jointly optimal congestion control and power control," *IEEE Journal on Selected Areas in Communications*, vol. 23, no. 1, pp. 104–116, 2005.

[18] J. F. Bonnans and A. Shapiro, *Perturbation Analysis of Optimization Problems*, Springer Series in Operations Research, Springer, New York, NY, USA, 2000.

[19] D. G. Luenberger, *Introduction to Linear and Nonlinear Programming*, Addison-Wesley, Reading, Mass, USA, 1973.

[20] W. I. Zangwill, *Nonlinear Programming: A Unified Approach*, Prentice-Hall, Englewood Cliffs, NJ, USA, 1969.