Hindawi Publishing Corporation
EURASIP Journal on Wireless Communications and Networking
Volume 2009, Article ID 878419, 15 pages
doi:10.1155/2009/878419

Research Article

On Connectivity Limits in Ad Hoc Networks with Beamforming Antennas

Moritz Kiese, Christian Hartmann, Julian Lamberty, and Robert Vilzmann

Institute of Communication Networks, Technische Universität München, Arcisstraße 21, 80290 München, Germany

Correspondence should be addressed to Christian Hartmann, hartmann@tum.de

Received 2 December 2008; Revised 10 June 2009; Accepted 19 August 2009

Recommended by Wolfgang Utschick

Multihop wireless networks will play an important role in future communication networks. Beamforming antennas have shown great potential to improve the connectivity in these networks; however only heuristic approaches exist to assess the benefits of this technology. Using the popular “keyhole” antenna model, we formulate a Mixed Integer Program (MIP) capable of acquiring optimal antenna configurations for path probability with various auxiliary constraints, node degree, and k-connectivity. We employ a problem-specific large-scale optimization approach named “column generation” to be able to cope with realistic scenario sizes without sacrificing optimality. In a case study, we show the feasibility of our optimization approach and demonstrate that “smarter” beamforming heuristics would still have plenty of room to improve the connectivity of such a network.

1. Introduction

The concept of multihop wireless communication is steadily gaining importance and will be an integral part of future wireless heterogeneous networking architectures. In this paper we consider multihop ad hoc and sensor networks, which operate without the help of any central entity in a distributed manner. Network nodes can either communicate via a direct link, if the two nodes are within communication range, or via multihop, utilizing intermediate nodes as relays. If the network topology does not allow for a path between two specific nodes in the network, there will be no way for these nodes to communicate, regardless whatever clever routing or data forwarding algorithms are applied. Hence, connectivity is of major importance in ad hoc or sensor networks. Optimally, there exists a multihop path between any node pair in the network. In this case the network is said to be *connected*. However, due to the random nature of the network topology in ad hoc networks, full connectivity is not always achieved. In this case the degree of connectivity can be expressed by the *path probability*, the probability that any selected pair of nodes in the network can set up a multihop path between them. For a random network topology, this probability depends on a number of variables: the spatial positions of the nodes, the transmit power, the channel path loss, the transmit and receive antenna gains, and the required receive power for successful decoding. If all these parameters are fixed, the connectivity of the given network cannot be improved. However, if we assume that some of the described parameters can be adapted, we can employ this to improve the network connectivity. This approach is called *connectivity shaping*. The most obvious parameter to change is the transmit power. This has been investigated in the past and has shown interesting gains. This connectivity shaping approach however inherits some limitations. Obviously, it would be beneficial to allow each node to choose an individual transmit power, that is locally optimal. Unfortunately, this will yield directed links, which is extremely undesired and will complicate both routing as well as medium access significantly. Hence, all nodes should agree on one common transmit power value. This does not only limit the degree of freedom for connectivity shaping, it also imposes signaling and protocol requirements, assuring the network wide use of the same power value.

In this paper, we consider a different approach to connectivity shaping. We fix the transmit power, but we adapt the transmit and receive antenna gains. In order to be able do this, the nodes must be equipped with adaptive antennas, that is, with an antenna array and adequate signal processing capabilities. By adapting the complex antenna gains appropriately, the main part of the power can be radiated in a desired direction, while radiation into other directions is significantly attenuated. Therefore, we can without increasing the transmit power increase the range of a node in some directions while decreasing the range in others. Obviously, this can change the connectivity of the network significantly. The options for receive and transmit beamforming are manyfold. Depending on the number of antennas and the available information (direction of arrival from other network nodes), several different goals can be achieved by means of beamforming: range maximization, interference minimization, SINR maximization, nulling of specific directions, to name a few. In the context of ad hoc networks, we are especially interested in beamforming approaches, which do not require frequent measurements, estimations, or information updates on a packet basis. We are rather interested in (semi-)static beamforming for connectivity shaping. Specifically, we have considered maximum gain phase shift beamforming; that is, the phase shift between the array antennas is chosen such that the resulting gain in a desired direction is maximized. Now, given the number of antennas and the maximum gain beamforming approach, the question arises, in which direction should any node in the network steer their maximum power, that is, their main beam, in order to maximize the path probability?

Several distributed approaches have been investigated. In the *Random Direction Beamforming* (RDB) approach, each node chooses a random direction and it could be shown that this simple approach already achieves significantly higher connectivity than omnidirectional networks with the same transmit power. The Penrose theorem, which graph theoretically links the network connectivity with the minimum node degree in random graphs and in geometric random graphs, motivated the *Maximum Node Degree Beamforming* (MNDB) approach. In this approach, each node performs beam sweeping, overhearing beacons, and selects the main beam direction, in which it identifies the maximum number of direct neighbors. This approach can in fact further increase the connectivity. The next logical step has been the *Two-Hop Neighbor Degree Beamforming* (TNDB), which maximizes the two-hop neighbors of each node locally, also based on beam sweeping and the exchange of beacons. This approach is further improving the connectivity. In parallel, other distributed beamforming strategies have been suggested in the literature. Performance comparison among the distributed approaches can easily be made by computer simulations. However, one important question remained unanswered so far: How far away are the distributed approaches developed up to now compared to the global optimum? Answering this question is extremely important in order to assess the quality of distributed algorithms and in order to evaluate the potential of developing improved distributed schemes. To this end, we develop a *Mixed Integer Linear Program* (MIP) formulation. This is enabled by applying a simplified yet meaningful smart antenna model, namely, the *brick wall* or *key hole* model. With the given optimization model, we are able to incorporate different optimization goals, such that on top of the path probability, other network properties can be optimized. To enable fast convergence of the optimization when the number of network nodes is increased, a column generation approach is used. With this optimization methodology, we are now able to identify performance limits of ad hoc networks in terms of connectivity, which can be applied in order to assess current and future distributed connectivity shaping approaches.

The reminder of the paper is organized as follows. We present the previous work in the field of ad hoc connectivity and Linear Programming in Section 2. Our underlying network model is introduced in Section 3. Based on this foundation, we develop our optimization model comprised of the MIP formulation and problem-specific solution methods in Section 4. In the following case study in Section 5, we compare a heuristic method for connectivity shaping to optimal results generated with our MIP model and draw our conclusions in Section 6.

2. Related Work

In this section, we discuss related literature regarding ad hoc network connectivity and linear optimization, respectively.

2.1. Connectivity of Ad Hoc Networks. Connectivity in ad hoc networks has been studied analytically and by means of system simulations. There is a body of research concentrating on omnidirectional nodes, relating the statistical connectivity to the node density and transmission range, for example, [1–4]. Most studies focus on homogeneous spatial node distributions, however, some investigate inhomogeneous distributions. The analytical studies on connectivity are mostly applying a theorem by Penrose [5] in order to relate the connectivity to the minimum node degree in the network. Penrose investigated random graph topologies and was able to prove that, under certain circumstances, the network becomes connected with high probability when the minimum node degree in the graph becomes one. Similarly, he could show that k-connectivity is achieved with high probability, when the minimum node degree becomes k [6]. Connectivity shaping by means of power control has been proposed and investigated, for example, in [7–11]. Connectivity studies considering adaptive beamforming antennas are not so common but do exist [12–14]. Adaptive antennas have been considered for connectivity shaping in [15, 16] by applying heuristic distributed approaches. To the best of our knowledge, an approach to determine connectivity bounds for ad hoc networks with adaptive antennas by means of optimization has not been provided so far by other authors. However, parts of the work presented in this paper have been published before in [17–19].

In contrast to these results, in this paper we consider additional optimization goals on top of the path probability maximization, like node degree maximization or minimization of the required capacity. Additionally, we provide a formulation and results for k-connectivity. Furthermore, the applied methods have been extended (including column generation) in order to be able to solve the more complex optimization problems within reasonable computational time for networks with a significant number of nodes.

2.2. Linear Optimization. *Linear Optimization* is a relatively young field. Although Fourier studied linear inequalities at the beginning of the 19th century, *Linear Programs* (LPs) began to attract widespread attention with the development of the *Simplex Method* by Dantzig in 1947 to solve military planning problems. (A verbose introduction to Linear Programs and the basic Simplex Algorithms can be found in [20]. More recent introductory texts are in [21, 22] while [23] offers a comprehensive overview of the field). A further boost in popularity was fueled by the availability of well performing and easy to use LP solvers such as ILOG CPLEX [24].

In addition to pure LP solvers, implementations of branch-and-bound algorithms to solve LPs with integer variables, so-called *Integer Linear Programs* (ILPs) or *Mixed Integer Programs* (MIPs), have become available, with ILOG CPLEX being again a high-performance commercial solution and SCIP developed by the Zuse Institute Berlin [25] offering more flexibility for advanced solution methods, like column generation or branch-and-cut.

The use of LPs in the context of network planning problems has a long and both scientific as well as commercially successful history (for an overview of standard network planning problems, formulations, and solution methods, we refer to [26]). Outstanding examples are capacity planning for fixed-networks [27] and radio network planning in GSM networks [28, 29]. Due to the sheer size of real-life problems (the necessary number of variables and constraints), the application of problem-specific solution algorithms such as *column generation* [30], *Bender's decomposition* [26, page 192ff], and *branch-and-cut* [31] gained considerable attention in the optimization community.

One of the most important properties of MIPs is information about the solution quality; that is, an *optimal* solution is a provable optimum in the mathematical sense. However, if an optimum could not be found (e.g.; due to time limitations), a solution *gap* between valid lower and upper bounds would be provided. This feature makes MIPs an extremely valuable tool for benchmarking heuristics and gaining provable knowledge about network properties as we will show in this paper.

3. Network Model

This section covers our network model which comprises the spatial node distribution, the wireless channel model, the link model, and the adaptive antenna model. Additionally, we briefly discuss connectivity measures and their relevance.

3.1. Node Placement. We consider random network topologies, generated by placing n nodes randomly in a two-dimensional system area of size $(1000\,\text{m})^2$. We are specifically interested in inhomogeneous node distributions because they are more realistic than homogeneous distributions and also more challenging with respect to connectivity. For this purpose, we first determine five cluster centers. The positions of the cluster centers are randomly chosen from a uniform distribution within the system plane. The n nodes are then split evenly among the cluster centers. Finally the node positions are chosen according to a spatial Gaussian distribution relative to the respective cluster center. The standard deviation of the Gaussian distribution is 10% of the area length.

3.2. Wireless Channel. We describe the wireless channel by the distance dependent path gain between transmit and receive antennas. We do not consider fast or shadow fading. Therefore, we assume a modified free space path loss model with attenuation exponent a. Thus, the received power at the receiver n_2 of a given link l_{n_1,n_2} can be expressed as

$$p_\text{r} = p_\text{t} \cdot g(\gamma_{n_1}) \cdot g(\gamma_{n_2}) \cdot \left(\frac{\lambda}{4\pi \cdot d_{n_1,n_2}}\right)^2 \left(\frac{d_0}{d_{n_1,n_2}}\right)^{a-2}, \quad (1)$$

where p_t is the transmission power, λ is the carrier wavelength (center frequency of the band used for transmission), d_{n_1,n_2} is the distance between n_1 and n_2, and d_0 is a reference distance. The factor $g(\gamma_{n_1})$ is the antenna gain of the transmitter n_1 in the direction of the receiver n_2, and $g(\gamma_{n_2})$ is the antenna gain of n_2 in the direction of n_1.

With the reference distance of $d_0 = 1\,\text{m}$, the received signal power p_r can now be written as

$$p_\text{r} = p_\text{t} \cdot g(\gamma_{n_1}) \cdot g(\gamma_{n_2}) \cdot \left(\frac{\lambda}{4\pi}\right)^2 \left(\frac{1}{d_{n_1,n_2}}\right)^{a} \quad (2)$$

for readability reasons, we ommit the 1/m multiplications necessary to remove the unit metres.

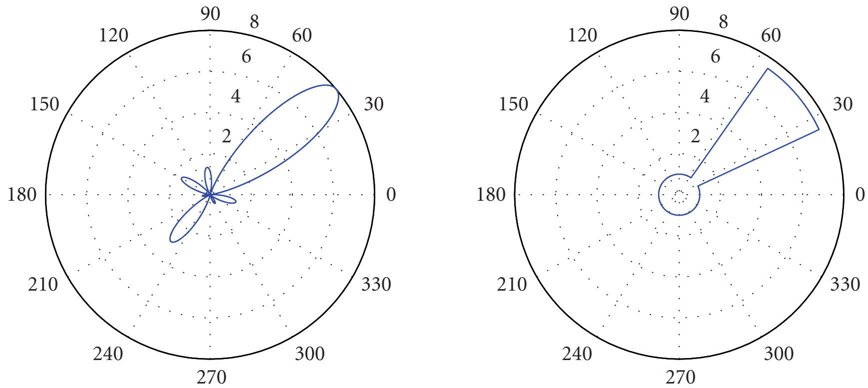
Moving to the dB-domain, the received signal power level becomes

$$P_\text{r} = P_\text{t} + G(\gamma_{n_1}) + G(\gamma_{n_2}) + 10 \cdot \log\left(\frac{\lambda}{4\pi}\right)^2 + 10 \cdot \log\left(d_{n_1,n_2}^{-a}\right), \quad (3)$$

with upper case variables denoting logarithmic scale. For the numerical results provided in this paper the attenuation exponent is chosen as $a = 3$.

3.3. Adaptive Antenna. We assume that each node in our network is equipped with an array of antennas, which can be used for adaptive beamforming. Specifically, we consider *Uniform Circular Array* (UCA) antenna configurations and *maximum gain phase-shift beamforming*; that is, the phase shift between neighboring antenna elements is chosen such as to maximize the antenna gain in a given direction. This type of beamforming applied to a UCA typically produces a single mainlobe and a number of sidelobes with significantly lower gains. Figure 1(a) illustrates the antenna gain of such a phased array with a circular arrangement of eight antenna elements, as a function of the angle in a two-dimensional plane. The antenna elements are modeled as ideal isotropic point radiators.

The resulting beamforming patterns are appropriate for simulative analysis. However, in this work it is our goal to optimize the connectivity in our network and for this purpose we are striving to formulate the problem as a Mixed



(a) Realistic pattern (b) Keyhole pattern

Figure 1: Antenna patterns.

Integer Linear Program (MIP). This would be extremely difficult for antenna patterns as in Figure 1(a).

Therefore, we require an antenna model, which is simpler and more abstract but still reflects the main characteristics of a realistic antenna pattern. A perfect fit to these requirements is the so-called *keyhole model*, which is commonly used in the literature (e.g., by [32]). Within this model, the gain pattern is described by an angular range with high antenna gain G_{M} (main lobe), and a low antenna gain G_{S} in the remaining directions (Figure 1(b)). G_{S} models the side lobes of a realistic antenna.

It can be argued that the fixed sidelobe level of the keyhole pattern does not reflect the fact that realistic beam-patterns have distinct nulls between the lobes. This means that in a real system communication in this null directions will hardly be possible, even if the nodes are very close. A way to acknowledge this within the usage of the keyhole pattern would be to set the sidelobe gain to zero. However, this seems to be too extreme since in this case two nodes can only establish a link, if both main lobes are steered towards each other. A more moderate approach is to choose the sidelobe gain G_{S} low enough (however, not zero in linear notation), such that links will hardly be established between nodes which are both steering away from each other, even if they are close together. We choose to apply the latter approach. Throughout this paper, we assume $G_{\mathrm{M}} = 10 = 10\,\mathrm{dBi}$ within an antenna aperture of $\alpha = 30^\circ$, and $G_{\mathrm{S}} = 0.1 = -10\,\mathrm{dBi}$ outside the aperture. (It should be noted that Figure 1(b) depicts an example pattern with different parameters than used for our investigations, just for illustration of the typical keyhole shape.) With those parameters, links between nodes which are both steering away from each other can only be established if the nodes are very close and we find from our numerical results that these cases are very rare. The fact that we are not completely eliminating this case, however, seems to be justified. From simulative investigations of connectivity with realistic beamforming we find that sometimes indeed links are established through sidelobes on either side (see, e.g., [13]).

3.4. Wireless Link. In order to determine the connectivity in a given network, we assume that a signal can be decoded, and thus a link can be established, if the received power P_{r} exceeds a receiver-dependent minimum signal power level $P_{\mathrm{r,min}}$. Therefore, two nodes will be within communication range if the following link budget inequation is fulfilled:

$$0 \le P_{\mathrm{t}} + G(\gamma_{n_1}) + G(\gamma_{n_2}) - P_{L,n_1,n_2} - P_{\mathrm{r,min}}. \qquad (4)$$

In (4) we used the logarithmic path loss, which is defined as

$$P_{L,n_1,n_2} = -10 \cdot \log\left(\frac{\lambda}{4\pi}\right)^2 - 10 \cdot \log\left(d_{n_1,n_2}^{-a}\right). \qquad (5)$$

With (4) and (5) we can now express the maximum communication range d_{r} for given angles γ_{n_1}, γ_{n_2} and respective antenna gains $G(\gamma_{n_1})$, $G(\gamma_{n_2})$ as

$$d_{\mathrm{r}} = 10^{(1/10\cdot a)\cdot\left(-P_{\mathrm{r,min}}+P_{\mathrm{t}}+G(\gamma_{n_1})+G(\gamma_{n_2})+10\cdot\log(\lambda/4\pi)^2\right)}. \qquad (6)$$

Using this expression, we calculate the largest possible distance $d_{\max}$ of two nodes, which are still able to communicate (both keyholes pointing to each other)

$$d_{\max} = 10^{1/10\cdot a\cdot\left(-P_{\mathrm{r,min}}+P_{\mathrm{t}}+2G_{\mathrm{M}}+10\cdot\log(\lambda/4\pi)^2\right)}, \qquad (7)$$

whereas two nodes being not farther away than $d_{\min}$ will be always able to communicate according to our keyhole model

$$d_{\min} = 10^{1/10\cdot a\cdot\left(-P_{\mathrm{r,min}}+P_{\mathrm{t}}+2G_{\mathrm{S}}+10\cdot\log(\lambda/4\pi)^2\right)}. \qquad (8)$$

In our network model, we assume that all nodes apply the same antenna pattern for both transmission and reception. Together with the channel model, this yields symmetric, that is, bidirectional links in our system.

3.5. Connectivity. The focus of our work is on the connectivity of the network. The connectivity is an important property because it describes the *reachability* of the nodes in the network. A classical communication network will always be planned such that each node pair can communicate, that is, is connected. Ad hoc networks cannot be planned and therefore full connectivity cannot be guaranteed; consequently some or many node pairs cannot communicate. Obviously we would like to minimize the number of node pairs that are unable to communicate by means of connectivity shaping. An appropriate measure for the connectivity of a given network is the path probability P_{path}. It is defined as the probability that any randomly chosen node pair in the network can establish a multihop path between them. In a network of N nodes the path probability is computed as

$$P_{\text{path}} = \frac{2n_{\text{path}}}{N(N-1)}, \tag{9}$$

where n_{path} is the number of node pairs in the network that can establish a multihop path between them. The network is connected if and only if $P_{\text{path}} = 1$. The goal of connectivity shaping is to maximize P_{path}. Once we have shaped the connectivity of our network, common MAC and routing protocols can be applied, operating on top of the given connectivity graph. Since we are dealing with connectivity shaping only, interference does not play a role in our model. Parallel transmissions generating interference in the network will depend on traffic demands at a certain time instant and the respective protocols for MAC and routing. However, regardless which type of communication protocols will be applied, a maximized path probability will always be desired, as discussed above.

However, often it will be possible to connect the network ($P_{\text{path}} = 1$) with many different realizations of antenna directions. In these cases we have the freedom to optimize additional objectives on top. Specifically, we will discuss three additional properties which will be incorporated in our optimization model as described in Section 4:

(a) maximal average node degree,

(b) minimal average node degree,

(c) minimal average path length (in terms of hop counts).

These objectives are not as generally useful as the path probability. Therefore, we will discuss briefly under which traffic assumptions which of them can be beneficial. (a) Maximal average node degree: in a low load scenario with large packet interarrival times, interference will not be a major issue, since parallel transmissions in closeby nodes are unlikely. In this case high node degrees can help against path breaks. Additionally, high node degrees can help broadcast messages to be flooded through the network with fewer steps and thus faster. (b) Minimal average node degree: in a high load scenario interference plays a major role. Each transmission will cause major interference at each direct neighbor, amounting to high interference levels at each node. Under this assumption, a lower node degree seems to be beneficial. (c) Minimal average path length: in a homogeneous traffic scenario, in which we have traffic demand for each node pair with a similar likelyhood, minimizing the average number of hops will be beneficial for end-to-end delay. It also minimizes the average number of required channel uses (necessity to access the channel), which will help to reduce the number of failed MAC attempts due to collisions on the medium. The minimal average path

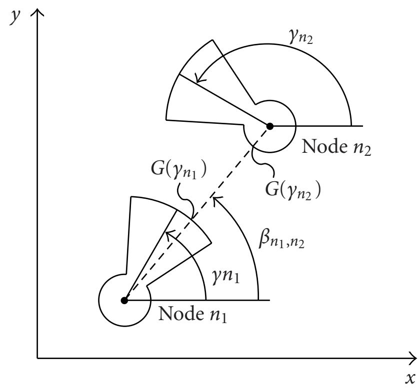


Figure 2: Example configuration.

length is achieved within the optimization framework by minimizing the *used capacity*, which will be introduced as part of the optimization model in Section 4.

4. Optimization Model

4.1. Link Indication. Inspecting (3) and (4) closely shows that the existence of a link between two nodes depends on the distance d_{n_1,n_2} between them and their antenna orientations γ_{n_1} and γ_{n_2} in relation to the angle β_{n_1,n_2} between them. Figure 2 shows an example configuration. We have to distinguish three cases.

4.1.1. Always in Range. If two nodes are so close that, regardless of the orientation of their main lobe, a link between them will exist, we can set $l_{n_1,n_2} = 1$ without further computation ($d_{n_1,n_2} < d_{\min}$). This will be the case if and only if d_{n_1,n_2} is so small that even the low antenna gain G_S at both nodes is sufficient to compensate for the path loss between them.

4.1.2. Out of Range. The other extreme is two nodes being so far apart that, not even by pointing their main lobes at each other and thus amplifying the signal with the highest possible gain, the signal power level at the receiver P_r will exceed $P_{r,\min}$ ($d_{n_1,n_2} > d_{\max}$). In this case, we can set the according link variables to zero or simply drop them completely from our problem formulation.

4.1.3. In Range in Case of Proper Steering. The most important case to take into consideration for our optimization problem will occur, if the existence of a link depends on the steering γ of the main lobes of the two nodes ($d_{\min} \le d_{n_1,n_2} \le d_{\max}$). Given the aperture α, the locations of the two nodes, and the angle β_{n_1,n_2} between them, we have to distinguish three cases for which the signal is amplified by the high gain G_M at node n_1 due to the angle discontinuity at 0 and 2π. Therefore, $G(\gamma_{n_1}) = G_M$ holds, if and only if

(i) for $\alpha/2 \le \beta_{n_1,n_2} \le 2\pi - \alpha/2$,

$$\gamma_{n_1} \ge \beta_{n_1,n_2} - \frac{\alpha}{2} \bigwedge \gamma_{n_1} \le \beta_{n_1,n_2} + \frac{\alpha}{2}, \tag{10}$$

(ii) for $2\pi - \alpha/2 \le \beta_{n_1,n_2} \le 2\pi$,

$$\gamma_{n_1} \ge \beta_{n_1,n_2} - \frac{\alpha}{2} \bigvee \gamma_{n_1} \le \beta_{n_1,n_2} - 2\pi + \frac{\alpha}{2}, \quad (11)$$

(iii) for $0 \le \beta_{n_1,n_2} \le \alpha/2$,

$$\gamma_{n_1} \ge \beta_{n_1,n_2} - \frac{\alpha}{2} + 2\pi \bigvee \gamma_{n_1} \le \beta_{n_1,n_2} + \frac{\alpha}{2}. \quad (12)$$

Using the first case (10) as an example, we will show in the following how to set up a set of inequalities to formulate a conditional link budget with binary indicator variables which finally depend on the antenna orientation γ. For notational convenience, we construct a graph $\mathcal{G}(\mathcal{N}, \mathcal{E})$, where an edge $e \in \mathcal{E}$ exists for every possible link between any node-pair $e := \{n_1, n_2\} \subset \mathcal{N}$ (i.e., the first and the last case). For the optimization problem, we choose binary link variables l_e to indicate whether a link between n_1 and n_2 is established ($l_e = 1$) or not ($l_e = 0$). Since we assume symmetric antenna configurations (i.e., sending and receiving lobe cannot be steered individually), it is sufficient to consider undirected edges.

First, we introduce two continuous variables $a'_{n_1,n_2} \in]-\beta_{n_1,n_2} + \alpha/2 + 1, 2\pi - \beta_{n_1,n_2} + \alpha/2 + 1]$ and $a''_{n_1,n_2} \in]-2\pi + \beta_{n1,n2} + \alpha/2 + 1, \beta_{n1,n2} + \alpha/2 + 1]$ for the first and second part of (10), respectively. Using the following equalities, $a'_{n_1,n_2} \ge 1$, indicates that we meet the first part of (10), whereas $a''_{n_1,n_2} \ge 1$ indicates the same for the second part

$$\begin{aligned} a'_{n_1,n_2} &= \gamma_{n_1} - \beta_{n_1,n_2} + \frac{\alpha}{2} + 1, \\ a''_{n_1,n_2} &= -\gamma_{n1} + \beta_{n1,n2} + \frac{\alpha}{2} + 1. \end{aligned} \quad (13)$$

a'_{n_1,n_2} and a''_{n_1,n_2} will then be used to set two *binary* variables b'_{n_1,n_2} and b''_{n_1,n_2} to zero, if $a'_{n_1,n_2} \ge 1$ and $a''_{n_1,n_2} \ge 1$, respectively. This is achieved by the following constraints. Obviously this could have been done in one single step without the use of the variables a'_{n_1,n_2} and a''_{n_1,n_2}. We will use these variables as a notational convenience to influence the link setup. The performance penalty for our MIP is negligible, because even simple presolvers are able to change the necessary expressions:

$$\begin{aligned} \left(-\beta_{n_1,n_2} + \frac{\alpha}{2}\right) \cdot b'_{n_1,n_2} - a'_{n_1,n_2} + 1 \le 0, \\ \left(-2\pi + \beta_{n_1,n_2} + \frac{\alpha}{2}\right) \cdot b''_{n_1,n_2} - a''_{n_1,n_2} + 1 \le 0. \end{aligned} \quad (14)$$

Equation (10) requires both of its inequalities to be fulfilled at the same time. If this is the case, the main lobe of n_1 will point into the direction of n_2, which we once more indicate by a binary variable b_{n_1,n_2}. It will become 1 in the following constraint, if and only if both inequalities are fulfilled, that is, $b'_{n_1,n_2} = b''_{n_1,n_2} = 0$, and zero otherwise

$$b_{n_1,n_2} \le \frac{1}{2} \cdot \left(1 - b'_{n_1,n_2}\right) + \frac{1}{2} \cdot \left(1 - b''_{n_1,n_2}\right). \quad (15)$$

In a similar way, we formulate constraints for the inequalities (11) and (12) which are stated in the appendix. Finally, this allows for a conditional formulation of the link budget (4):

$$\begin{aligned} &(P_t - P_{r,\min} - P_{L,n_1,n_2} + 2 \cdot G_S) l_e \\ &\quad + b_{n_1,n_2} \cdot (G_M - G_S) + b_{n_2,n_1} \cdot (G_M - G_S) \ge 0. \end{aligned} \quad (16)$$

This constraint describes the possible configurations of a node pair by either taking the high gain G_M or the low gain G_S at the nodes into account, depending on the orientation of their main lobes. Consequently, the binary link indicator variable l_e for a connection between n_1 and n_2 can only be set to 1, if the link budget is sufficient and has to be zero otherwise.

Note that in the end, the value of the l_e's and subsequently the associated indicator variables determine the steering γ of the main beam of the nodes. Our model was developed in a bottom-up manner here. In contrast, during the upcoming optimization, the values of γ will be determined top-down, by requiring certain links to be established in the solution.

4.2. Optimization Goals. In this section, we will formulate a variety of optimization goals on top of the previously described network model.

4.2.1. Node Degree. One of the first applications of the optimization model developed above was a benchmark for the MNDB heuristic [17]. As this heuristic tries to maximize the node degree, our first cost function is a simple maximization of the number of links

$$\max \sum_{e \in \mathcal{E}} l_e. \quad (17)$$

4.2.2. Connectivity. One of the most obvious questions arising from a given scenario as described in Section 3.1 is whether a steering configuration of the beamforming antennas exists, so that the network is connected. Despite the simplicity of the question, this cannot be answered via readily available algorithms. Furthermore, in cases where full connectivity is not possible, the highest *path probability* is of similar interest.

In order to answer this question, we introduce a demand set $\mathcal{D}$ with a demand $D_d = 1$, for all $d \in \mathcal{D}$ for every node pair $\{n_s, n_t\} \subset \mathcal{N}$ (i.e., a fully meshed demand matrix). Using a node-link flow-formulation, we use flow-variables $f^d_{n_1,n_2} \in \mathbb{R}^+_0$ denoting the flow for a given demand d on a link between nodes n_1 and n_2. Quite obviously, this very link has to be existing; that is, the link-indicator value l_e from the previous section has to be 1

$$\sum_{d \in \mathcal{D}} f^d_{n_1,n_2} \le l_e \cdot |\mathcal{D}|, \quad \forall \{n_1, n_2\} \subset \mathcal{N}. \quad (18)$$

A demand d will only be satisfied, if and only if the sum of the outgoing flows from the source node of the demand n_s is as large as the demand D_d:

$$\sum_{n_2 \in \mathcal{N}_{n_s}} f^d_{n_s,n_2} = D_d, \quad \forall d \in \mathcal{D}, \tag{19a}$$

where $\mathcal{N}_{n_s}$ represents the set of adjacent nodes of n_s. Similarly, the sum of all incoming flows for the demand d at the target node n_t has to be as large as D_d as well

$$\sum_{n_1 \in \mathcal{N}_{n_t}} f^d_{n_1,n_t} = D_d, \quad \forall d \in \mathcal{D}. \tag{19b}$$

In any other (intermediate) node n, all incoming flows have to be as large as the outgoing flows for this demand (flow conservation), which follows the philosophy of Kirchhoff's current law, namely,

$$\sum_{n_1 \in \mathcal{N}_n} f^d_{n_1,n} - \sum_{n_2 \in \mathcal{N}_n} f^d_{n,n_2} = 0, \quad \forall n \in \mathcal{N} \setminus \{n_s, n_t\}, \forall d \in \mathcal{D}. \tag{19c}$$

It is worth noting that the constraints above can only be fulfilled if and only if the network can be connected. Since this can be by no means guaranteed, we keep the set of constraints feasible, by adding a so-called *dummy variable* $\delta_d \in \mathbb{R}^+_0$ to the sink and source constraints, which consequently are transformed to

$$\sum_{n_2 \in \mathcal{N}_{n_s}} f^d_{n_s,n_2} + \delta_d = D_d, \quad \forall d \in \mathcal{D}, \tag{20a}$$

$$\sum_{n_1 \in \mathcal{N}_{n_t}} f^d_{n_1,n_t} + \delta_d = D_d, \qquad \forall d \in \mathcal{D}. \tag{20b}$$

No changes to the flow conservation constraint (19c) are necessary, because zero-valued flow variables will always meet this constraint. Our objective is to satisfy as many demands as possible without using the dummy variables δ_d

$$\min \sum_{d \in \mathcal{D}} \delta_d. \tag{21}$$

Solving the presented MIP now causes the MIP solver to minimize the number of dummy variables with nonzero values, that is, by using the flow-variables. However, the flow-variables $f^d_{n_1,n_2}$ can be used if and only if the associated link-indicator variables l_e are set to 1, which in return requires a suitable antenna configuration.

An interesting possibility of this formulation is, to gain a k-connected steering configuration, by imposing an upper limit of 1 to the flow-variables ($f^d_{n_1,n_2} \in [0;1]$) and increasing the demand per node pair D_d to k.

While the presented formulation will gain the desired result (a network with the highest possible path probability), we are able to improve the performance (i.e., solution time and memory consumption) of the formulation notably by reducing the number of variables and constraints. For our purposes it is not necessary to differentiate between demands originating from one source node n_s, hence we simplify our demand set $\mathcal{D}$, by aggregating the demands for one source n_s, that is, we have a $d \in \mathcal{D}'$ for every node $n_s \in \mathcal{N}$ and $D_d = |\mathcal{N}| - 1$. It is worth noting however, that it is not sufficient to generate demands from one single node to all remaining nodes, since this one node could reside within an island. In this situation the resulting antenna configuration would only affect nodes reachable within the same island. With this modification we only need one source constraint per source node, namely,

$$\sum_{n_2 \in \mathcal{N}_{n_s}} f^d_{n_s,n_2} + \delta_d = D_d, \quad \forall d \in \mathcal{D}. \tag{22a}$$

However, since all nodes in $\mathcal{N}$ are also sinks, the flow conservation and target node constraint are merged to

$$\sum_{n_1 \in \mathcal{N}_n} f^d_{n_1,n} - \sum_{n_2 \in \mathcal{N}_n} f^d_{n,n_2} + \delta_d = 1, \quad \forall n \in \mathcal{N} \setminus \{n_s\}, \forall d \in \mathcal{D}, \tag{22b}$$

which means that a demand of 1 has to end at every node (with the exception of n_s) for every demand d.

Connectivity and Auxiliary Constraints. Close inspection of the computational results of the MIP formulated above reveals that in many of our scenarios, a larger number of configurations exist, which share the same degree of connectivity. Consequently, we can formulate additional optimization goals to further differentiate among these solutions.

(a) Used Capacity Minimization. We introduce a new variable $u_{n_1,n_2} \in \mathbb{R}^+_0$ which will measure the capacity necessary on a link between n_1 and n_2, if we add the following constraint:

$$\sum_{d \in \mathcal{D}} f^d_{n_1,n_2} \leq u_{n_1,n_2}, \quad \forall \{n_1, n_2\} \subset \mathcal{N} \tag{23}$$

to our MIP. Extending the objective function by the total used capacity and weighting the dummy variables with C

$$\min\left(C \cdot \sum_{d \in \mathcal{D}} \delta_d + \sum_{\{n_1,n_2\} \subset \mathcal{N}} u_{n_1,n_2}\right) \tag{24}$$

will minimize the number of used dummy variables (i.e., maximizing the path probability) and then minimize the used capacity in the network if C is chosen arbitrarily high. An obvious minimal value of C would be, for example, the number of links in the network; that is, using a capacity of 1 on a dummy variable would be as expensive as using the same capacity on *every* link in the network. Since all our established links offer sufficient capacity to accommodate all demands, demands will flow along the shortest possible path to consume as little capacity as possible (i.e., 1 on every edge their path traverses). Consequently, we minimize the average shortest path length as discussed in Section 3.5. Naturally, a solution of this MIP will also be a valid solution for the previous "pure" model.

(b) Number of Links. Depending on the specific scenario Section 3.5, it could be desirable to either maximize the number of links in order to ensure a densely meshed network or to minimize the number of links (to reduce interference), which can be achieved by another extension of the objective function

$$\min\left(C \cdot \sum_{d\in\mathcal{D}} \delta_d + B \cdot \sum_{e\subset\mathcal{E}} l_e\right). \tag{25}$$

Again, we choose C arbitrarily high, so that the path probability remains the primary objective. Setting B to -1 will maximize the number of links while ensuring connectivity. In contrast, setting $B = 1$, the MIP solver will minimize the number of active links. Due to the use of inequalities hower, this is not sufficient to force the solver to steer keyholes away to prevent the creation of links.

4.3. Large-Scale Optimization Methods. During the addition of the auxiliary constraints however, performance problems of the node-link formulation became apparent. The large number of variables and constraints even in the more compact formulation in combination with the auxiliary constraints and relatively weak LP bounds makes the problem very hard to solve; in some cases we were not able to solve the 30-node scenario up to a reasonable optimization gap (e.g., 5%) within two days. In order to overcome this limitation we changed our formulation to the so-called *path-approach* which allows the use of problem-specific, large-scale optimization methods, such as in our case *column generation*.

4.3.1. Path-Flow Formulations. Instead of considering the node-link flows, we now introduce the notion of *path-flows* on top of our network model. A path-flow variable $f_{d,p} \in \mathbb{R}_0^+$ denotes the flow (i.e., amount of traffic) for a demand d along a path p. Satisfying a demand d therefore requires the flow along all paths $p \in \mathcal{P}_d$ to be at least as large as the demand D_d

$$[\pi_d] \qquad \sum_{p\in\mathcal{P}_d} f_{d,p} + \delta_d \geq D_d, \quad \forall d \in \mathcal{D}. \tag{26}$$

The demand set $\mathcal{D}$ is created similarly to the previous section, a fully-meshed demand set with $D_d = 1$, for all $d \in \mathcal{D}$. The used capacity constraint of paragraph Section 4.2.2(a) can be rewritten accordingly to

$$[\sigma_e] \qquad \sum_{d\in\mathcal{D}} \sum_{\substack{p\in\mathcal{P}_d:\\ e\in p}} f_{d,p} \leq u_e, \quad \forall e \in \mathcal{E}. \tag{27}$$

Again, in order to be usable for a path using a link from node n_1 to n_2, the corresponding link indicator variable l_e has to be 1

$$u_e \leq \mathcal{D} \cdot l_e, \quad \forall e \in \mathcal{E}. \tag{28}$$

The drawback of this approach is quite obvious. In order to guarantee optimality, we have to precompute all *possible* paths and create a path-flow variable $f_{d,p}$, which generates a huge number of variables.

4.3.2. Column Generation. Analysis of existing solutions however reveals that only a small fraction of all possible paths will be used in the solution; that is, only a very limited number of path-flow variables have nonzero values. As we can leave out zero vectors from our solution space, without losing optimality, we could start our optimization with exactly this (smaller) set of variables and would still obtain the same solution.

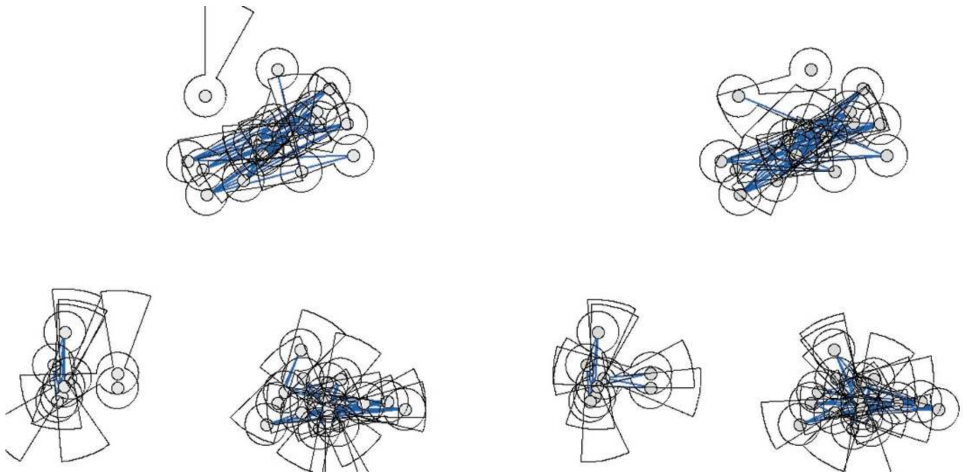
The mathematical methodology to exploit this situation appeared implicitly in the *Dantzig-Wolfe-Decomposition* [33]. This principle of *(delayed) column generation*. In the standard matrix-vector notation of MIPs, variables correspond to columns in the coefficient matrix, hence the name was explicitly applied in [34, 35] for the cutting-stock-problem. In cutting-stock problems, a single stock (paper, fabric, etc) which comes in one width has to be cut according to customer specifications with minimal waste.

In column generation the path-flow formulation above with *all* possible paths is called *Master Problem* (MP). We start our optimization with a *restricted master problem* (RMP), with a very limited number of path variables, the only requirement being, that the MIP is feasible (which in our case is true without any of the path-flow variables due to the dummy variables). Our task is now to find variables, or in other words paths, which will improve the current solution, that is, gain a "better" solution once we resolve the program with the new additional variables. We solve this so-called *pricing problem* by inspection of the *dual system*. The dual system can be easily constructed by using Lagrangean Relaxation [22, page 140ff]. For this, we associated every primal constraint with exactly one variable (denoted in angular brackets in (26) and (27)). Since every variable in the primal system (the RMP denoted above) is associated with exactly one constraint in the dual, we acquire the following constraints for the variables in question (namely, $f_{d,p}$):

$$\pi_d - \sum_{e\in p} \sigma_e \leq 0, \quad \forall d \in \mathcal{D}, \ \forall p \in \mathcal{P}_d. \tag{29}$$

Now every primal variable not (yet) in the solution base, which would improve the solution, will violate this dual constraint. We can easily acquire the value of the dual variables in most modern LP solvers without any additional overhead, because they employ a refinement of the original simplex algorithm, the *dual-simplex algorithm*. Hence our task of finding improving variables consists of finding those paths for every demand d, which would violate the previous constraint. Due to the structure of our MP, both $\pi_d \in \mathbb{R}_0^+$ as well as $\sigma_e \in \mathbb{R}_0^+$. With this, we can view our task as a shortest path problem: find the shortest (cheapest) path with link-weights σ_e. This problem can be solved in polynomial time by Dijkstra's algorithm [36]. If the total cost is smaller than π_d, this path will improve the solution, otherwise no improving variables for demand d exist. After this we will resolve the new RMP (with additional paths) and repeat the path search until no improving variables can be found any more, which means that we have found the optimal solution.

It is important to note however that our problem contains integer variables which implies that we have to



(a) MNDB (b) Maximum sum node degree

Figure 3: 40-node example network.

run the column generation algorithm in every node of the branch-and-bound tree [37], stressing the importance of an effective pricing algorithm.

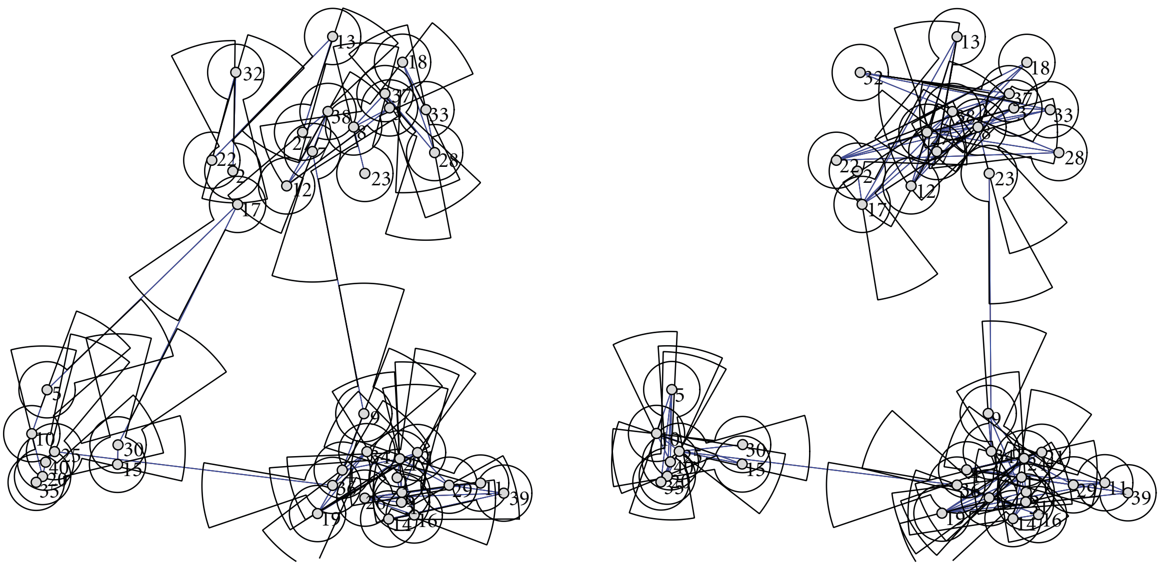
5. Case Study

In the following, solutions from the optimization will be depicted and an evaluation of the computational performance will be presented. The previously described model was implemented using SCIP 1.1 [25] with CPLEX 9.13 [24] as LP-solver backend. The same tools were employed for a flow-based approach [18] which was used as reference for the performance evaluation. We will discuss some distinct results for specific scenarios first. This can give some insight into the different properties of the discussed optimization objectives. Subsequently, we present numerical results on the path probability averaged over 100 random scenarios. In both cases we compare the results of the optimization to a distributed heuristic, namely, *Maximum Node Degree Beamforming* [15]. Three specific node scenarios are considered which differ in the number of nodes: $n = \{30, 40, 50\}$. Each scenario is randomly generated as described in Section 3.1.

5.1. Node Degree Maximization. Let us begin with the maximization of the sum node degree. The motivation for the sum node degree optimization was to find a bound which can be used to assess a specific heuristic beamforming approach, namely, *Maximum Node Degree Beamforming* (MNDB). The MNDB algorithm has been proposed in [15] and was developed to allow for connectivity shaping in a distributed decentralized manner. In MNDB, each network node starts by performing beam sweeping, determining the number of neighbors in each direction by means of overhearing periodical beacons. The sweeping is performed in discrete steps of γ_{sweep} degrees. Once the full 360° have been sampled, the node chooses the beamdirection, in which the maximum number of neighbors is seen. Obviously, MNDB attempts to maximize the local node degree. The idea behind MNDB was clearly to make use of the connection between the minimum node degree and the connectivity which could be proven for geometric and pure random graphs by Penrose in [5, 6]. For our 40-node example network, the result of MNDB is depicted in Figure 3(a). As a comparison, the network with maximized sum node degree is shown in Figure 3(b). It can be seen that even though the MNDB already provides good connectivity. In [15] it could be shown that MNDB in fact significantly improves the connectivity as compared to *Random Direction Beamforming* (RDB) and even more compared to omnidirectional antennas. The optimized network shows better connectivity. However, looking at the resulting connectivity in both cases in Figure 3, we observe disconnected clusters, which obviously is unfavorable.

5.2. Path Probability Maximization. While the node degree maximization is specifically interesting to assess MNDB, we should bear in mind that performance measures like connectivity or path probability are much more important for the functionality of an ad hoc network. Therefore, we are interested in optimized results for such properties, in order to be able to assess the performance of distributed realistic algorithms. The most important measure is the connectedness of the network. If the network is connected, all participants can communicate with each other. If a network, due to its node distribution cannot be fully connected, we prefer the path probability to be as high as possible. Therefore we choose path probability maximization to be our prime optimization criterion. For networks with a relatively high node density, many solutions exist that yield a connected network. This leaves room to include additional optimization goals that can further improve the network performance in practice. (Note that the results obtained from the optimization with additional criteria are valid for pure path probability optimization also.)

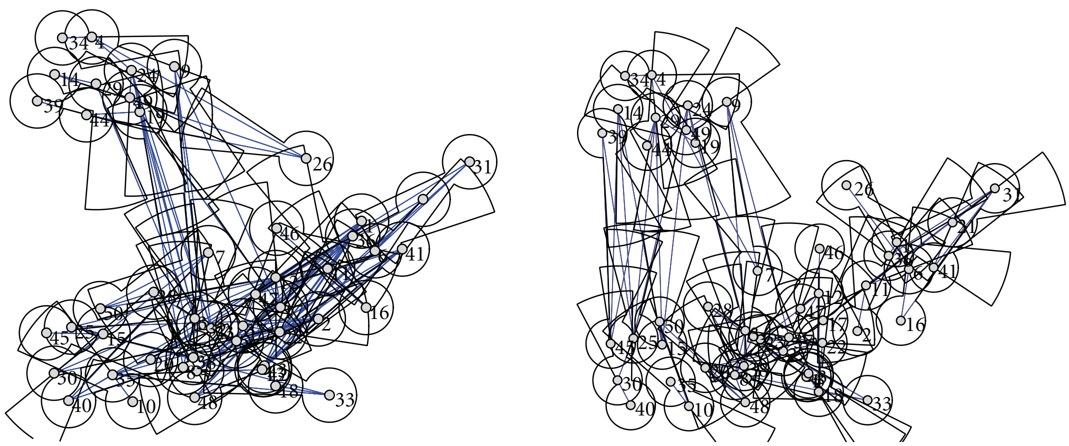
Figure 4(a) depicts an optimized result for the 40-node scenario which was achieved by path probability maximization. It can be seen that the resulting network is connected.



(a) Path probability optimization only, $d_{\text{avg}} = 2.35$

(b) Path probability optimization and node degree maximization, $d_{\text{avg}} = 3.80$

Figure 4: 40-node example network.



(a) Path probabilty optimization only

(b) Path probabilty optimization and used capacity minimization

Figure 5: 50-node example network.

5.2.1. Auxiliary Objectives. (a) *Node Degree Maximization.* The depicted solution in Figure 4(a), however, is just one of many solutions yielding full connectivity. A different solution achieving a connected network is shown in Figure 4(b). This network configuration is obtained, when in addition to the prime optimization goal of path probability maximization, the additional goal of node degree maximization is considered in the optimization model as described in Section 4.2.2(b). Note that node degree maximization comes at the expense of higher interference, but results in a network, which is more robust against node failures inside the clusters.

(b) *Capacity Minimization.* Another desirable objective in order to improve network throughput is the minimization of packet collision probability on the MAC level. This can be achieved by minimizing the used capacity in our optimization model following Section 4.2.2(a), which yields a connected network with minimum total hop count. This minimizes the number of channel access attempts and thus the MAC level collision probability. Figures 5(a) and 5(b) show the optimization results for a 50-node example network, comparing pure path probability maximization to used capacity minimization as additional objective.

(c) *2-Connected Network.* Finally, we present an optimization result for the 30-node network scenario. In this case the optimization objective is to achieve two edge-disjoint paths between any node pair in the network. As can be seen from Figure 6, this yields a 2-connected network, increasing the robustness.

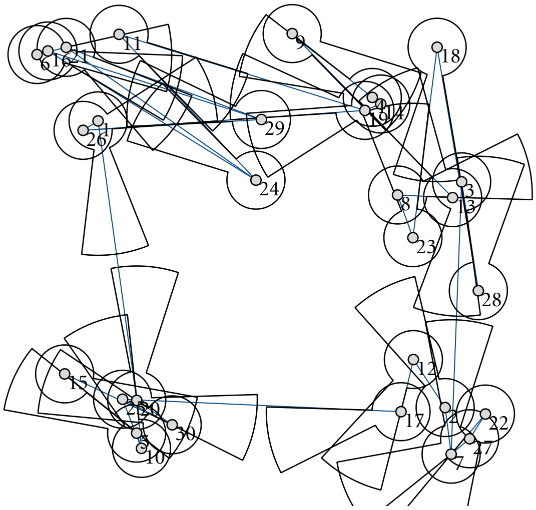


Figure 6: Path probability optimization for 2-connected network.

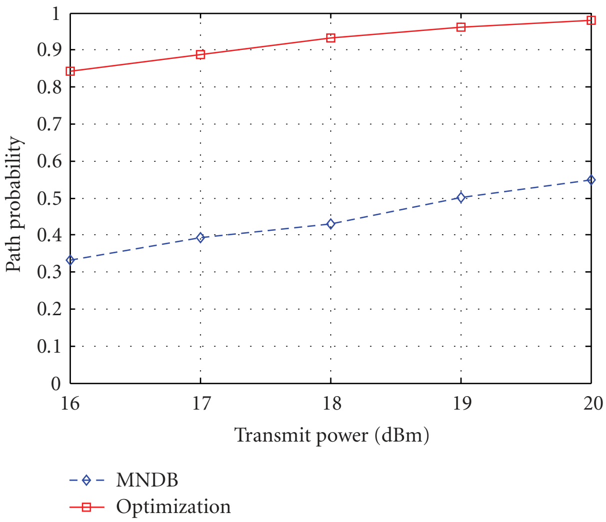


Figure 7: Path probability optimization versus MNDB.

5.3. Statistical Analysis. In addition to the specific examples discussed previously, we have also obtained numerical results for 100 different random scenarios, each with 30 nodes. We discuss the results with respect to the path probability, our prime objective. Figure 7 compares the path probability averaged over all scenarios obtained with MNDB and with optimization over the transmit power, which is varied between 16 dBm and 20 dBm in steps of 1 dBm.

In both cases the obtained path probability grows with increased transmit power, as expected. It can also be seen that the MNDB approach on average gives results which are far from the optimum. This is due to its limitation to local information and local measures (local node degree). Specifically in these relatively sparse scenarios with only 30 nodes, often clusters can only be connected if nodes from both clusters cooperate and steer their beams towards each other, as MNDB is not able to support systematically.

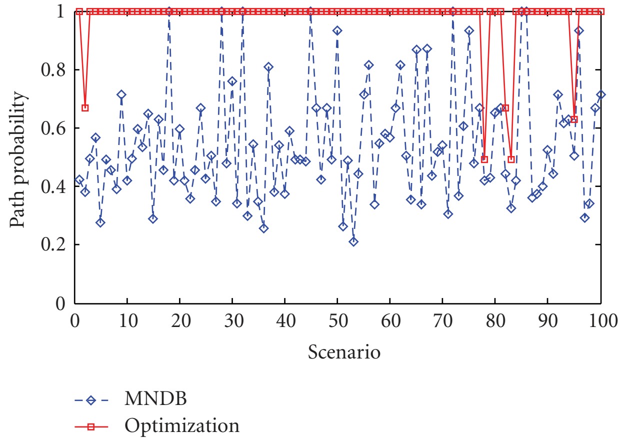


Figure 8: Path probability optimization versus MNDB per scenario, $P_t = 20$ dBm.

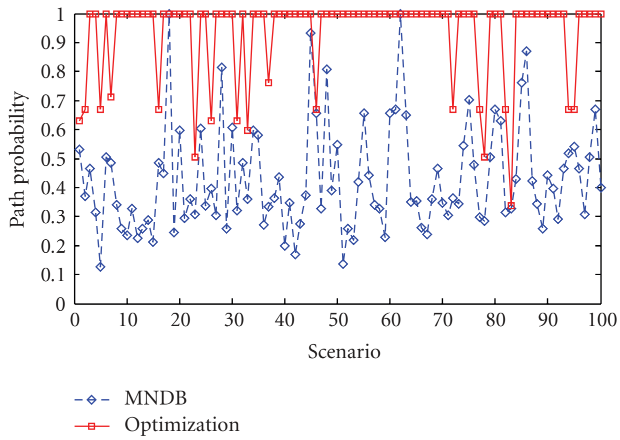


Figure 9: Path probability optimization versus MNDB per scenario, $P_t = 18$ dBm.

Figures 8 and 9 depict the path probability for both, optimization and MNDB for each scenario for the case of transmit powers of $P_t = 20$ dBm and $P_t = 18$ dBm, respectively. It can be observed that the achievable path probability heavily depends on the specific scenario realization. Also the gap between MNDB and the optimum varies strongly between the different scenarios. In fact, in some cases MNDB is able to generate a connected network, hence achieving the optimum.

From the 100 investigated random scenarios, we also generated CDF plots for the path probability. These plots additionally contain the results in terms of path probability in case the optimization model is used to maximize the average node degree, only. Inspecting Figure 10 ($P_t = 20$ dBm) and Figure 11 ($P_t = 18$ dBm), we observe a huge gap between path probability optimization and MNDB, encouraging additional research efforts for improved local heuristics. Even the optimization of the average node degree achieves a better path probability with respect to the MNDB heuristic. However, the performance gap between MNDB and average node degree optimization is relatively small.

Table 1: Performance data.

	30 nodes		40 nodes		50 nodes	
Path probability maximization	Flow approach	Path approach	Flow approach	Path approach	Flow approach	Path approach
Solving time	58 s	> 1440 min	2 min 19 s	> 1440 min	8 min 32 s	101 min 24 s
LP iterations	161 000	> 3 060 000	188 000	> 25 400 000	110 000	352 000
Maximum memory usage	89 MB	> 885 MB	143 MB	> 1670 MB	497 MB	158 MB
Node degree maximization						
Solving time	69 min 19 s	21 min 37 s	> 1440 min	69 min 34 s	> 1440 min	775 min 30 s
LP iterations	205 000 000	732 000	> 217 000 000	944 000	> 13 700 000	2 630 000
Maximum memory usage	171 MB	107 MB	> 9800 MB	182 MB	> 522 MB	296 MB
Used capacity minimization						
Solving time	n/a	> 1440 min	n/a	353 min 37 s	n/a	399 min 41 s
LP iterations	n/a	> 81 300 000	n/a	12 400 000	n/a	21 700 000
Maximum memory usage	n/a	> 453 MB	n/a	336 MB	n/a	287 MB

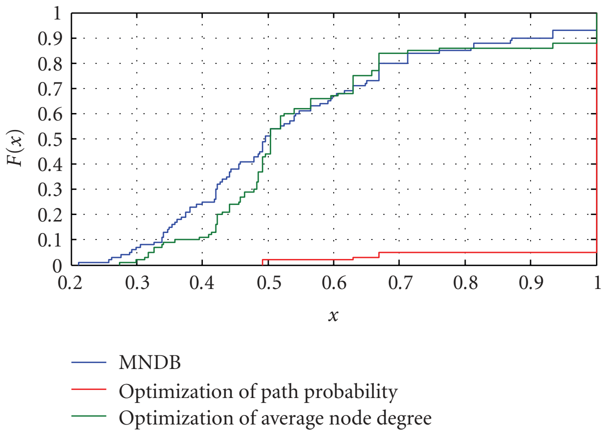


Figure 10: CDF of path probability, comparison of MNDB, optimization of path probability, and optimization of average node degree, $P_t = 20$ dBm.

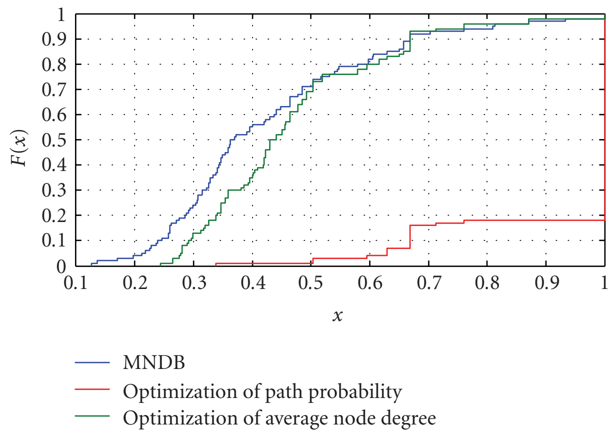


Figure 11: CDF of path probability, comparison of MNDB, optimization of path probability, and optimization of average node degree, $P_t = 18$ dBm.

From this we conclude that aiming at maximizing node degrees—even if done optimally and though motivated by Penrose's theorem as discussed above—will not be sufficient in realistic networks if we want to achieve good overall connectivity.

With the presented results, we have shown how we can achieve performance bounds for many different connectivity properties of ad hoc networks. These can serve as a means to assess distributed connectivity shaping algorithms with respect to various performance goals. The gap between the optimization and available or novel heuristics can indicate how much room there remains for improved heuristics. Our results show that in fact there is great potential for advanced heuristics. On the other hand, by inspecting and analysing optimized results for different networks, the optimization results can also be of help in developing better heuristics in the future.

5.4. Computational Performance of the Optimization. An evaluation of the computational complexity, in particular execution time and memory usage in Table 1 shows that for the path probability maximization the classic flow approach (as presented in [18]) performs better than a path approach with column generation. The latter was at its most impressive when it came to additional node degree maximization or used capacity minimization. Here, it outperforms the flow approach by far. The 40 nodes and 50 nodes example networks could not be solved to optimality with the flow approach within a time limit of 48 hours when additional objectives were applied. Note that we did not systematically evaluate the performance of the flow approach for the used capacity optimization, as tests did not perform very promising. Performance data was collected on a dual Intel Xeon X5260 machine with 16 GB of RAM.

6. Conclusion

We have provided an MIP formulation for the optimization of an array of connectivity criteria for ad hoc networks with adaptive antennas, by choosing the optimal beamdirections. Providing results for a sample of ad hoc network scenarios, we could show how much potential still remains for connectivity and performance improvements of ad hoc networks with adaptive antennas. We believe that these performance bounds can be very useful for the research community interested in further improving the merits of adaptive antennas in ad hoc networking. By applying a column generation approach, the computation time for finding the optimum could be significantly reduced, such that the optimization of larger networks can now be achieved in a reasonable amount of time, allowing for fast evaluation of many different network scenarios and objectives. Future work will focus on finding optimal beampatterns for ad hoc connectivity.

Appendix

Complete Formulation

In this section we provide the complete formulation up to the link indicator variables l_e. An edge $e \in \mathcal{E}$ exists for those node-pairs $\{n_1, n_2\} \subset \mathcal{N}$, where the distance d_{n_1,n_2} is smaller than $d_{\max}$

(i) $\forall e = \{n_1, n_2\} \in \mathcal{E} : d_{\min} \le d_{n_1,n_2} \le d_{\max} \bigwedge \alpha/2 \le \beta_{n_1,n_2} \le 2\pi - \alpha/2$,

$$\begin{aligned} a'_{n_1,n_2} &= -\beta_{n_1,n_2} + \frac{\alpha}{2} + 1 + \gamma_{n_1}, \\ a''_{n_1,n_2} &= \beta_{n_1,n_2} + \frac{\alpha}{2} + 1 - \gamma_{n_1}, \\ \left(-\beta_{n_1,n_2} + \frac{\alpha}{2}\right) &\cdot b'_{n_1,n_2} - a'_{n_1,n_2} + 1 \le 0, \\ \left(-2\pi + \beta_{n_1,n_2} + \frac{\alpha}{2}\right) &\cdot b''_{n_1,n_2} - a''_{n_1,n_2} + 1 \le 0, \\ \frac{1}{2} \cdot \left(1 - b'_{n_1,n_2}\right) + \frac{1}{2} &\cdot \left(1 - b''_{n_1,n_2}\right) - b_{n_1,n_2} \ge 0, \end{aligned} \tag{A.1}$$

(ii) $\forall e = \{n_1, n_2\} \in \mathcal{E} : d_{\min} \le d_{n_1,n_2} \le d_{\max} \bigwedge 2\pi - \alpha/2 \le \beta_{n_1,n_2} \le 2\pi$,

$$\begin{aligned} c'_{n_1,n_2} &= -\beta_{n_1,n_2} + \frac{\alpha}{2} + 1 + \gamma_{n_1}, \\ c''_{n_1,n_2} &= \beta_{n_1,n_2} - 2\pi + \frac{\alpha}{2} + 1 - \gamma_{n_1}, \\ \left(-\beta_{n_1,n_2} + \frac{\alpha}{2}\right) &\cdot b'_{n_1,n_2} - c'_{n_1,n_2} + 1 \le 0, \\ \left(-4\pi + \beta_{n_1,n_2} + \frac{\alpha}{2}\right) &\cdot b''_{n_1,n_2} - c''_{n_1,n_2} + 1 \le 0, \\ 1 - b'_{n_1,n_2} + 1 - b''_{n_1,n_2} &- b_{n_1,n_2} \ge 0, \end{aligned} \tag{A.2}$$

(iii) $\forall e = \{n_1, n_2\} \in \mathcal{E} : d_{\min} \le d_{n_1,n_2} \le d_{\max} \bigwedge 0 \le \beta_{n_1,n_2} \le \alpha/2$,

$$\begin{aligned} d'_{n_1,n_2} &= -\beta_{n_1,n_2} + \frac{\alpha}{2} - 2\pi + 1 + \gamma_{n_1}, \\ d''_{n_1,n_2} &= \beta_{n_1,n_2} + \frac{\alpha}{2} + 1 - \gamma_{n_1}, \\ \left(-2\pi - \beta_{n_1,n_2} - \frac{\alpha}{2}\right) &\cdot b'_{n_1,n_2} - d'_{n_1,n_2} + 1 \le 0, \\ \left(-2\pi + \beta_{n_1,n_2} + \frac{\alpha}{2}\right) &\cdot b''_{n_1,n_2} - d''_{n_1,n_2} + 1 \le 0, \\ 1 - b'_{n_1,n_2} + 1 - b''_{n_1,n_2} &- b_{n_1,n_2} \ge 0, \end{aligned} \tag{A.3}$$

(iv) $\forall e \in \mathcal{E} : d_{\min} \le d_{n_1,n_2} \le d_{\max}$,

$$\begin{aligned} &(P_{\mathrm{t}} - P_{\mathrm{r,min}} - P_{L,n_1,n_2} + 2 \cdot G_{\mathrm{S}}) l_e \\ &\quad + b_{n_1,n_2} \cdot (G_{\mathrm{M}} - G_{\mathrm{S}}) + b_{n_2,n_1} \cdot (G_{\mathrm{M}} - G_{\mathrm{S}}) \ge 0, \end{aligned} \tag{A.4}$$

(v) $\forall e = \{n_1, n_2\} \in \mathcal{E} : d_{n_1,n_2} \le d_{\min}$,

$$l_e = 1, \tag{A.5}$$

with

$$\begin{aligned} &\gamma_{n_1} \in [0, 2\pi] \subset \mathbb{R}, \\ &f_{d,p}, \delta_d \in [0, 1] \subset \mathbb{R}, \\ &u_e \in [0, H] \subset \mathbb{R}, \\ &l_e, b_{n_1,n_2}, b'_{n_1,n_2}, b''_{n_1,n_2} \in [0, 1] \subset \mathbb{N}, \\ &a'_{n_1,n_2} \in \left[-\beta_{n_1,n_2} + \frac{\alpha}{2} + 1, -\beta_{n_1,n_2} + \frac{\alpha}{2} + 1 + 2\pi\right], \\ &a''_{n_1,n_2} \in \left[\beta_{n_1,n_2} + \frac{\alpha}{2} + 1 - 2\pi, \beta_{n_1,n_2} + \frac{\alpha}{2} + 1\right] \subset \mathbb{R}, \\ &c'_{n_1,n_2} \in \left[-\beta_{n_1,n_2} + \frac{\alpha}{2} + 1, -\beta_{n_1,n_2} + \frac{\alpha}{2} + 1 + 2\pi\right] \subset \mathbb{R}, \\ &c''_{n_1,n_2} \in \left[-4\pi + \beta_{n_1,n_2} + \frac{\alpha}{2} + 1, \beta_{n_1,n_2} - 2\pi + \frac{\alpha}{2} + 1\right] \subset \mathbb{R}, \\ &d'_{n_1,n_2} \in \left[-\beta_{n_1,n_2} + \frac{\alpha}{2} - 2\pi + 1, -\beta_{n_1,n_2} + \frac{\alpha}{2} + 1\right] \subset \mathbb{R}, \\ &d''_{n_1,n_2} \in \left[\beta_{n_1,n_2} + \frac{\alpha}{2} - 2\pi + 1, \beta_{n_1,n_2} + \frac{\alpha}{2} + 1\right] \subset \mathbb{R}. \end{aligned} \tag{A.6}$$

Acknowledgments

The authors' gratitude goes to the SCIP developers, namely, Tobias Achterberg, Timo Berthold, and Kati Wolter for their strong support and instantaneous bug-fixing capabilities. They furthermore want to thank the anonymous reviewer who provided the insight to improve the applicability of their link-budget inequality.

References

[1] C. Bettstetter and C. Hartmann, "Connectivity of wireless multihop networks in a shadow fading environment," in *Proceedings of the 6th ACM International Workshop on Modeling, Analysis and Simulation of Wireless and Mobile Systems (MSWiM '03)*, pp. 28–32, ACM, New York, NY, USA, 2003.

[2] C. Bettstetter, "On the minimum node degree and connectivity of a wireless multihop network," in *Proceedings of the 3rd International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc '02)*, pp. 80–91, New York, NY, USA, June 2002.

[3] P. Santi, D. M. Blough, and F. Vainstein, "A probabilistic analysis for the range assignment problem in ad hoc networks," in *Proceedings of the 2nd ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc '01)*, pp. 212–220, New York, NY, USA, 2001.

[4] C. Bettstetter, *Mobility modeling, connectivity, and adaptive clustering in ad hoc networks*, Ph.D. dissertation, Technische Universität München, Munich, Germany, 2003.

[5] M. D. Penrose, "The longest edge of the random minimal spanning tree," *Annals of Applied Probability*, vol. 7, no. 2, pp. 340–361, 1997.

[6] M. D. Penrose, "On k-connectivity for a geometric random graph," *Random Structures and Algorithms*, vol. 15, no. 2, pp. 145–164, 1999.

[7] P. Santi and D. M. Blough, "The critical transmitting range for connectivity in sparse wireless ad hoc networks," *IEEE Transactions on Mobile Computing*, vol. 2, no. 1, pp. 25–39, 2003.

[8] X.-Y. Li, P.-J. Wan, Y. Wang, and C.-W. Yi, "Fault tolerant deployment and topology control in wireless networks," in *Proceedings of the International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc '03)*, pp. 117–128, Annapolis, Md, USA, June 2003.

[9] L. Wang, W. Zhao, Y. Li, Y. Qu, Z. Liu, and Q. Chen, "Sleep-supported and cone-based topology control method for wireless sensor networks," in *Proceedings of the IEEE International Conference on Networking, Sensing and Control (ICNSC '08)*, pp. 1445–1448, Sanya, China, April 2008.

[10] H. Liu, B. Zhang, J. Zheng, and H. T. Mouftah, "An energy-efficient localized topology control algorithm for wireless ad hoc and sensor networks," *International Journal of Communication Systems*, vol. 21, no. 11, pp. 1205–1220, 2008.

[11] D. M. Blough, M. Leoncini, G. Resta, and P. Santi, "Topology control with better radio models: implications for energy and multi-hop interference," *Performance Evaluation*, vol. 64, no. 5, pp. 379–398, 2007.

[12] R. Vilzmann, C. Bettstetter, D. Medina, and C. Hartmann, "Hop distances and flooding in multihop networks with randomized beamforming," in *Proceedings of the 8th ACM Symposium on Modeling, Analysis and Simulation of Wireless and Mobile Systems (MSWiM '05)*, pp. 20–27, Montreal, Canada, October 2005.

[13] C. Bettstetter, C. Hartmann, and C. Moser, "How does randomized beamforming improve the connectivity of ad hoc networks?" in *Proceedings of the IEEE International Conference on Communications (ICC '05)*, Seoul, Korea, May 2005.

[14] H. Koskinen, "Analytical study of connectivity in wireless multihop networks utilizing beamforming," in *Proceedings of the 9th ACM Symposium on Modeling, Analysis and Simulation of Wireless and Mobile Systems (MSWiM '06)*, pp. 212–218, New York, NY, USA, 2006.

[15] R. Vilzmann, J. Widmer, I. Aad, and C. Hartmann, "Low-complexity beamforming techniques for wireless multihop networks," in *Proceedings of the 3rd Annual IEEE Communications Society on Sensor and Adhoc Communications and Networks (SECON '06)*, vol. 2, pp. 489–497, Reston, Va, USA, September 2006.

[16] X. Zhou, H. M. Jones, S. Durrani, and A. Scott, "Effects of beamforming on the connectivity of ad hoc networks," in *Proceedings of the 8th Australian Communications Theory Workshop (AusCTW '07)*, Adelaide, Australia, February 2007.

[17] C. Hartmann, M. Kiese, and R. Vilzmann, "Optimizing the node degree in wireless multihop networks with single-lobe beamforming," in *Proceedings of the 9th INFORMS Telecommunications Conference (INFORMS '08)*, College Park, Md, USA, March 2008.

[18] M. Kiese, C. Hartmann, and R. Vilzmann, "Optimality bounds of the connectivity of adhoc networks with beamforming antennas," in *Proceedings of the IEEE GLOBECOM (GLOBECOM '09)*, Honolulu, Hawaii, USA, 2009.

[19] M. Kiese, C. Hartmann, and J. Lamberty, "Optimality criteria for the connectivity of adhoc networks with beamforming antennas," in *Proceedings of the International Network Optimization Conference (INOC '09)*, Pisa, Italy, April 2009.

[20] G. B. Dantzig, *Linear Programming and Extensions*, Princeton University Press, Princeton, NJ, USA, 1963.

[21] V. Chvátal, *Linear Programming*, W. H. Freeman, New York, NY, USA, 1983.

[22] D. Bertsimas and J. N. Tsitsiklis, *Introduction to Linear Optimization*, Athena Scientific Optimization and Computation Series, Athena Scientific, Cambridge, Mass, USA, 1st edition, 1997.

[23] G. L. Nemhauser and L. A. Wolsey, *Integer and Combinatorial Optimisation*, John Wiley & Sons, Atlanta, Ga, USA, 1st edition, 1998.

[24] "ILOG CPLEX 9.0 User's Manual," ILOG, S. A., 2003.

[25] T. Achterberg, *Constraint integer programming*, Ph.D. dissertation, Technische Universität Berlin, Berlin, Germany, 2007, http://opus.kobv.de/tuberlin/volltexte/2007/1611.

[26] M. Pioro and D. Mehdi, *Routing, Flow, and Capacity Design in Communication and Computer Networks*, Elsevier, Amsterdam, The Netherlands, 2004.

[27] K. Ambs, S. Cwilich, M. Deng, D. J. Houck, D. F. Lynch, and D. Yan, "Optimizing restoration capacity in the AT&T network," *Interfaces*, vol. 30, no. 1, pp. 26–44, 2000.

[28] U. Türke, *Efficient methods for W-CDMA radio network planning and optimization*, Ph.D. dissertation, 2007.

[29] H.-F. Geerdes, *UMTS radio network planning: mastering cell coupling for capacity optimization*, Ph.D. dissertation, Technische Universität Berlin, Berlin, Germany, 2008.

[30] C. Gruber and M. Kiese, "Optimization of resilient networks with column generation," in *Proceedings of the 8th INFORMS Telecommunications Conference*, Dallas, Tex, USA, March 2006.

[31] A. M. C. A. Koster, S. Orlowski, C. Raack, G. Baier, and T. Engel, "Single-layer cuts for multi-layer network design problems," in *Telecommunications Modeling, Policy, and Technology*, vol. 44, chapter 1, pp. 1–23, Springer, College Park, Md, USA, February 2008.

[32] R. Ramanathan, "On the performance of ad hoc networks with beamforming antennas," in *Proceedings of the 2nd ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc '01)*, pp. 95–105, New York, NY, USA, 2001.

[33] G. B. Dantzig and P. Wolfe, “Decomposition principle for linear programs,” *Operations Research*, vol. 8, no. 1, pp. 101–111, 1960.

[34] P. C. Gilmore and R. E. Gomory, “A linear programming approach to the cutting-stock problem,” *Operations Research*, vol. 9, no. 6, pp. 849–859, 1963.

[35] P. C. Gilmore and R. E. Gomory, “A linear programming approach to the cutting stock problem part II,” *Operations Research*, vol. 11, no. 6, pp. 863–888, 1963.

[36] E. W. Dijkstra, “A note on two problems in connexion with graphs,” *Numerische Mathematik*, vol. 1, no. 1, pp. 269–271, 1959.

[37] C. Barnhart, E. L. Johnson, G. L. Nemhauser, M. W. P. Savelsbergh, and P. H. Vance, “Branch-and-price: column generation for solving huge integer programs,” *Operations Research*, vol. 46, no. 3, pp. 316–329, 1998.