Decomposition for Low-complexity Near-Optimal Routing in Multi-hop Wireless Networks

Vinay Kolar
Department of Wireless Networks,
RWTH Aachen University, Germany.
vko@mobnets.rwth-aachen.de

Nael B. Abu-Ghazaleh
Dept. of Computer Science,
State University of New York, Binghamton.
nael@cs.binghamton.edu

Petri Mähönen
Department of Wireless Networks,
RWTH Aachen University, Germany.
pma@mobnets.rwth-aachen.de

***Abstract*—Network flow models serve as a popular mathematical framework for the analysis and optimization of Multi-hop Wireless Networks. They also serve to provide the understanding necessary to derive effective distributed protocols. However, the high computational complexity of realistic models restrict the translation of theoretical insights into distributed protocols. In this paper, we consider an NP-hard, Mixed Integer Linear Programming based routing model that computes single-path routes in a wireless network. We propose an efficient, polynomial time algorithm that applies domain specific heuristics to reduce the complexity. We employ a decomposition based approach to break the monolithic problem into several sub-problems that cooperate to find near-optimal routes. The sub-problem structure is chosen such that it captures the optimal route discovery process between a source and destination; this is a design principle that can be directly used in distributed routing protocols. We show that the resulting formulation achieves orders of magnitude improvement in the run-time. Simulation results show that the routes derived from the model are effective even in practical wireless networks with commonly used protocol stack.**

I. Introduction

Ad hoc networks, mesh networks, and wireless sensor networks are instances of Multi-Hop Wireless Networks (MHWNs), in which nodes cooperate to forward traffic among each other. The available capacity to a pair of directly communicating nodes is influenced not only by the nominal channel capacity, but also by ongoing communication by nearby nodes because of the shared nature of the medium. More specifically, other ongoing transmissions make the medium busy, preventing transmission, or cause collisions wasting channel time. Despite these interactions between connections, the majority of routing algorithms route connections greedily, taking local decisions without any coordination. Typically, decisions are made for each connection considering metrics such as shortest path or the expected packet errors on a link; such policies may lead to routing connections to mutually interfering nodes when, perhaps, other regions of the network are idle.

One approach to alleviating this problem is to coordinate between the connections (traffic engineering) to reach an optimal configuration with respect to some desired objective function. Recently there has been a wave of research in deriving optimal protocols based on mathematical optimization problems [1]. Several efforts for modeling MHWNs within such a framework have been undertaken [2]–[6]. These models are largely network flow based, and take into account interference. This line of research is extremely promising for a number of reasons, including: (1) it provides practical limits for performance of specific MHWN networks, allowing more effective provisioning and design of these networks; (2) similarly, the developed models can be used to guide traffic engineering decisions to achieve effective usage of the available network resources; and (3) it provides understanding of the shape of effective solutions, and insight into what makes them effective, to aid in the design of protocols that converge to such solutions.

In this paper, we consider network-flow based routing models in MHWNs that can be more directly used in static MHWNs [5], [6]. They capture metrics that are directly measurable at high time granularity (e.g. busy-time), thus reducing the control information exchange overhead, and compute a set of single-path routes that can be directly provided as an input to the routing layer of MHWNs. The developed models are expressed as a Mixed Integer Linear Programming (MILP) problem [7], which is then solved relative to an objective function such as minimizing end-to-end delay or maximizing throughput. However, MILP problems are NP-hard. The MHWN problem is especially difficult because of the large number of constraints that result from mutual interference on the wireless channel among nearby nodes. In order to make the model more realistic, the effect of scheduling and the channel should be taken into consideration, further increasing the complexity of the problem. Thus, while the approach is promising, it is critically limited by the computational complexity, which grows exponentially with the size of the network. This limitation restricts the utility of the approach to small networks, and for off-line analysis.

We explore efficient solutions to the MILP formulation of MHWNs in two directions. First, we approximate the NP-hard problem by a polynomial time algorithm by eliminating the complex components using domain-specific heuristics. The lower complexity allows scaling the size of networks and helps to evaluate the effectiveness of the non-greedy routing approaches in moderate to large networks. We show that large reductions in run-time are achieved (up to 3 orders of magnitude in the scenarios we study); this gap is likely to increase as the models scale due to the much lower asymptotic complexity of the proposed formulation. Further, the quality of the routes obtained compares favorably with those obtained from the full model.

Secondly, we adopt the *decomposition* approach where the NP-hard monolithic optimization problem is broken down into smaller, more manageable, sub-problems. In our case, each sub-problem attempts to route an individual connection which abstracts the behavior of the discovery and maintenance of a route by the routing protocol. A master problem coordinates the sub-problems by monitoring the interference metrics in the neighborhood of the link, thus providing the abstraction of control information flow between the nodes of the network to enable routing decisions. These abstractions aid in the design of near-optimal routing protocols that are derived from a theoretical model.

The rest of the paper is organized as follows: Related work is discussed in Section II. Section III describes the general MILP based routing framework. The proposed formulation is explained in Section IV. Section V evaluates its run-time and performance benefits. Discussion and future work are outlined in Section VI. Finally, Section VII concludes the paper.

II. Related work

Interference-aware routing formulation in MHWN has been modelled in many studies [2]–[6]. Certain models, like [2]–[4], solve joint scheduling and routing problem using a fluid model of the data network where infinitesimally small amount of data is optimally scheduled at appropriate links during each time-slot. One of the primary goals of such studies is to maximize the theoretical capacity of the network. However, it is hard to apply the insights directly in practical MHWNs since: (1) the network operates under non-ideal and distributed scheduling protocols; (2) time-granularity for exchange of control information is much larger than a time-slot; and (3) packets cannot be split into infinitesimally smaller chunks.

Other research studies capture the effect of interference at a higher level and use this information to derive near-optimal routing configurations; e.g. Kolar *et al.* [5], [6] capture the busy-time observed at a node to calculate the optimal routing configuration. While these studies do not provide exact capacity estimates of the network, they focus on obtaining a near-optimal routing configuration that can be directly used in the network. For example, they explicitly restrict the effect of path-splitting to obtain a single-route per connection since a majority of the routing protocols do not support this feature due to the adverse effects like packet re-ordering. They also discuss and provide approximations for obtaining a fair-routing configuration where the objective function is sensitive to optimize all the connections and the interaction between the connections. Routing model with such features enable the deployment of efficient traffic-engineering applications on realistic networks. For example, optimal routes from such scenarios can be used in static mesh network backbone for provisioning and QoS applications. Since such models can be realized in a majority of commodity wireless networks, we aim to reduce the complexity and decompose such routing models in this paper.

Several authors have proposed heuristic routing protocols with different metrics to improve the performance of the routing in MHWNs [8], [9]. Our paper is different from this set of research work since we approach the problem from a modeling perspective in an attempt to find the optimal routing configuration and protocol behavior.

III. General routing model

We use as the starting point in this paper our previously developed routing framework [5], [6], henceforth called the *basic model*. The routing model is formulated as a Multi-commodity Flow (MCF) problem [10] where each connection is modelled as a "flow" and various performance parameters of the connection, like throughput [5] and end-to-end delay [6], are optimized. For the purpose of illustration, we consider a simple model that minimizes the end-to-end delay of the connections.

Consider a static MHWN that is represented as a graph $G(\mathcal{N}, \mathcal{E})$ where $\mathcal{N}$ denotes the set of nodes and $\mathcal{E}$ represents the set of edges between the nodes that can communicate. The set $\mathcal{E}$ can be chosen based on the distance or measured signal quality between the pair of nodes. Let (s_k, d_k, t_k) denote the source, destination and time required to transmit one packet for the k^{th} connection. Let $\mathcal{C}$ be the set of connections. The number of nodes, edges and connections are denoted by N, E and C, respectively. The set of links that interfere *with* link (i, j) is given by $\mathcal{W}_{ij}$ and the set of links that are interfered *by* link (i, j) is denoted by $\mathcal{B}_{ij}$. The amount of time dedicated by link (i, j) to carry one packet of connection k is denoted by the variable x_{ij}^k. The basic constraints are:

$$0 \le x_{ij}^k \le t_k \quad \forall k \in \mathcal{C}, \quad \forall (i,j) \in \mathcal{E} \quad (1)$$

$$b_i^k = \sum_{(i,j)\in\mathcal{E}} x_{ij}^k - \sum_{(j,i)\in\mathcal{E}} x_{ji}^k \quad \forall k \in \mathcal{C}, \quad \forall i \in \mathcal{N} \quad (2)$$

For a given connection k, the maximum amount of time to forward a single packet on any edge is t_k (Equation 1). The *demand* for a given node is the difference between the total outflow from the node and total inflow to the node. Equation 2 specifies the demand requirement to be met at each node.

A single single-path route is desirable to avoid multi-path routing [11] overheads; this constraint is represented by:

$$x_{ij}^k = t_k \cdot 1_{ij}^k \quad \forall k \in \mathcal{C}, \quad \forall (i,j) \in \mathcal{E} \quad (3)$$

where 1_{ij}^k is an indicator variable set to 1 if the edge carries the traffic for the $k\mathrm{th}$ connection and 0 otherwise. This variable transforms the MCF problem to an NP-hard *Mixed Integer Linear Program (MILP)* optimization problem [10].

Effective routing configurations are obtained by solving the above MCF problem against an appropriate objective function. The objective function is chosen such that the end-to-end delay is minimized. *Commitment period* A_{ij}^m of the link (i, j) for connection m is defined as the amount of time a link is busy in either accommodating interfering traffic or self-traffic and is given by

$$A_{ij}^m = 1_{ij}^m \Bigg(\sum_{k \in C} x_{ij}^k + \sum_{(y,z) \in \mathcal{W}_{ij}, k \in \mathcal{C}} x_{yz}^k \Bigg). \tag{4}$$

Commitment period (A_{ij}^m) is applicable for only the active links of connection m and is 0 if the link does not participate in the connection. This metric indicates the channel busy-time experienced by a link; an experimentally measurable heuristic that captures the effect of interference levels and which is chosen by several protocols (e.g. [9]).

The general form of the objective function is a weighted function of commitment periods of all the links and is given by:

$$\text{minimize} \sum_{m \in \mathcal{C}} p_m \sum_{(i,j) \in \mathcal{E}} A_{ij}^m. \tag{5}$$

The above equation minimizes the end-to-end commitment period for each connection m with a constant priority p_m assigned to connection m.

IV. A Decomposition based Formulation

An optimization problem can be decomposed using several approaches [1]. In this paper, we restructure the problem using a primal decomposition approach, where several smaller sub-problems solve a part of the optimization problem and a master problem coordinates by providing the necessary parameters to the sub-problems. We formulate the model such that each sub-problem provides an abstraction of an end-to-end connection set-up and the master problem is responsible for updating and disseminating the routing metric that are used by the routing protocol. We do not claim that this is the optimal or the only way of decomposing the above problem. We adopt this approach since it gives the right abstraction level and facility to control end-to-end routes through a distributed routing protocol.

We now reorganize the objective function of the primal problem such that the abstraction of the sub-problems are manifested. We then formulate the sub-problem where appropriate objective function and the constraints are chosen for the sub-problem. The formulation of the master problem which controls the sub-problems is then discussed. During these formulations we state the assumptions and approximations that were made to enable a polynomial time solution to the problem while preserving the practical constraints (e.g. a single-path route per connection).

A. Reorganizing the primal problem

The objective function in Equation 5 can be re-structured as:

$$\text{minimize} \sum_{k \in C} \sum_{(i,j) \in \mathcal{E}} (c_{ij}^{k^*} + s_{ij}^{k^*}) x_{ij}^k. \tag{6}$$

The weight for an edge (i,j) is split into the cross-connection and self-connection interference weights. Let n_{ij}^m denote the number of *active* links in connection m that interfere with (i,j). The cross-connection weight

$$c_{ij}^{k^*} = \sum_{m \in \mathcal{C}, m \neq k} p_m (1_{ij}^m + n_{ij}^m) \tag{7}$$

for the edge (i,j) for any connection k is the weighted additional busy-period introduced to the end-to-end delays of the other connections by choosing the edge (i,j). Similarly, the self-interference weight

$$s_{ij}^{k^*} = p_k (1_{ij}^k + n_{ij}^k) \tag{8}$$

is the busy-period introduced to the active links of the same connection. This result is derived in the Appendix. The decomposition approach we adopt is to split the objective functions into minimizing $\sum_{(i,j)\in\mathcal{E}} (c_{ij}^{k^*} + s_{ij}^{k^*}) x_{ij}^k$ in each sub-problem. The master problem assigns the appropriate value of the weight $c_{ij}^{k^*}$ for each edge (i,j) and connection k.

B. Formulation of the subproblem

The k^{th} sub-problem solves optimal path problem for the k^{th} connection. The bounds (Equation 1) and the flow constraints (Equation 2) are independent for each sub-problem and hence can be isolated for the k^{th} connection. The master problem, which is aware of the routing configurations chosen by the sub-problems, assigns appropriate weights to each link (c_{ij}^k). The sub-problem chooses the route by solving the optimization problem:

$$\text{minimize} \sum_{(i,j) \in \mathcal{E}} (c_{ij}^k + s_{ij}^k) x_{ij}^k \quad ,$$

such that:

$$\begin{aligned} 0 \leq x_{ij}^k &\leq t_k & \forall (i,j) \in \mathcal{E} \\ b_i^k = \sum_{(i,j) \in \mathcal{E}} x_{ij}^k &- \sum_{(j,i) \in \mathcal{E}} x_{ji}^k & \forall i \in \mathcal{N}. \end{aligned} \tag{9}$$

Inclusion of the self-interference effect will introduce the integer variable 1_{ij}^k and thus making the problem NP-hard. We argue that self-interference effect can be ignored due to two practical reasons. First, self-interference in a chain is inevitable whereas cross-connection interference can be avoided. Second, the purpose of the a polynomial time decomposition algorithm is to enable routing in larger networks with a number of connections. Hence, the effect of cross-interference is larger. Based on the above two reasons and in the interest of deriving a single single-path route in polynomial time, we do not consider the effect of self-interference (i.e. $s_{ij}^k = 0$). An approach to refine this estimate is to assign appropriate values of s_{ij}^k based on the routes observed by the master problem in previous iterations. We will investigate such techniques in the future and do not pursue such an approach in this paper.

In the resulting non-integer linear program, split routes solutions are still theoretically possible. However, we observe that split routes can be easily avoided when there are no upper bounds on the variable x_{ij}^k (refer to pages 379–381 in [12]). This is true in our case since the problem attempts to minimize the end-to-end commitment period without any hard upper bound on the delay.

The sub-problem with the above assumptions reduces into a single-source single-sink shortest path problem which can be solved by efficient algorithms like Dijkstra's algorithm, which can be solved for sparse graphs in $O(E + N\log(N))$ [10].

C. The Master Problem

The master problem is responsible for invoking the sub-problem with appropriate cross-connection link weights c_{ij}^k and ensuring convergence of the problem. However, exact computation of these weights are hard since each connection k needs the knowledge of active links of all the other connections m (n_{ij}^m of Equation 7) before solving the optimal routes for connection k. This prevents the clear splitting of the monolithic optimization problem into smaller sub-problems and complicates the attainment of global minimum. We employ heuristics in order to overcome this circular dependency.

The master problem iteratively invokes all the sub-problems while updating the values of c_{ij}^k. During each iteration, the updated c_{ij}^k forces the sub-problems to choose routes which have the least effect on other connections. As a result, the overall solution moves towards an effective solution with respect to the global objective function. A connection is considered solved if it repeats a solution for a preset number of times. The overall problem is considered to be solved when all the sub-problems are solved. For each invocation of the k^{th} sub-problem, the master problem updates cross-connection link weights c_{ij}^k and stores the current routes. The complexity of this book-keeping is $O(e_a E h_k)$ where e_a is average number of interferers for an edge (which depends upon the density of nodes in the network) and h_k is the number of hops in the k^{th} connection (which is bounded and small, and hence can be ignored).

Convergence: The convergence of the sub-problems is not guaranteed due to the coupling between the connections (n_{ij}^m of Equation 7). We now discuss the reasons that prevent convergence and propose rules to ensure faster and near-optimal convergence.

A sub-problem considers only the cross-interference experienced from other connections. This may lead to cyclic dependency between the routes chosen by the sub-problems. *Route-flapping* may occur where a sub-problem recurrently chooses a sequence of routes during consecutive iterations depending upon the routes chosen by the other connections. We observe such interactions and apply heuristics, like comparing with the previously computed routes, to detect flapping and force convergence.

We introduce a maximum number of iterations to avoid rare cases where sub-problems oscillate between a large number of routes. Finally, the order of execution of the sub-problems may lead the solution to be stuck at a local minima. In order to avoid this, we invoke the sub-problems in random order.

With the above heuristics, our simulation results show that the solution converges very fast; 95% of the random scenarios converged in 3-5 iterations. As we show in Section V, the quality of the routes obtained are also superior. Other mechanisms that allow escaping from local minima are topics we intend to examine more detail in the future.

V. Performance Evaluation

In this section, we first evaluate the effect of complexity reduction by comparing the run-time of proposed formulation with the basic model. We then verify the effectiveness of the theoretical ideas in practical networks by evaluating the performance of the formulation in MHWNs that use IEEE 802.11 and a dynamic routing protocol using QualNet simulator [13]. We show that the quality of the routes obtained from the decomposed problem are superior to the standard protocols and comparable to those of the basic model. We then demonstrate the scalability and performance benefits of non-greedy routing strategy in large random networks.

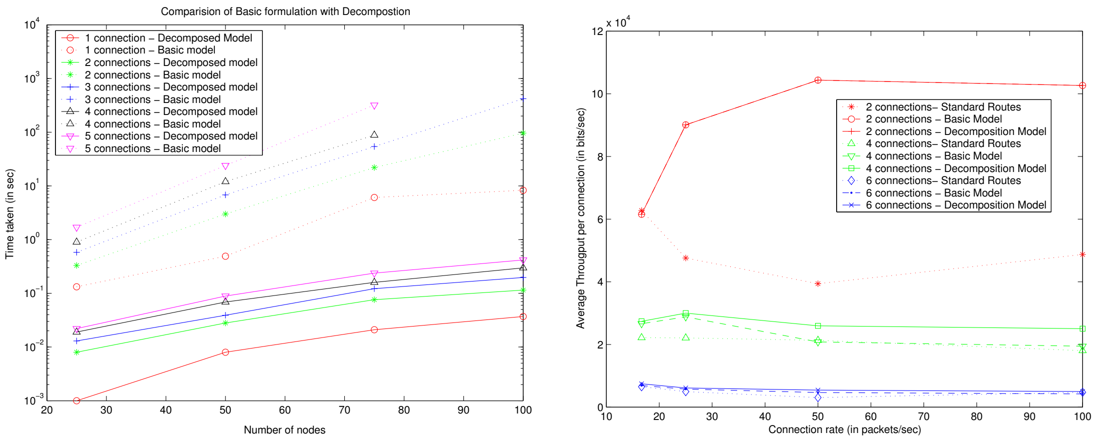
Run-time analysis was conducted by varying nodes and connections in a random network scenario. Each point represents the average of ten different networks with uniformly distributed nodes and connected pairs. The models were run on a standard desktop PC with 2 GHz Intel Pentium 4 processor and 512 MB RAM. Figure 1(a) illustrates that the decomposed problem is orders of magnitude faster to solve than the basic model. For example, an improvement of over 1000 *times* is seen at larger node and connection scenarios. Since the basic formulation is NP-hard, while the decomposed formulation polynomial, this advantage increases as the size of the problem scales.

Figure 1(b) compares the quality of the solution (in terms of throughput) obtained by a standard routing protocol[1], basic formulation and the decomposed formulation for a 6×6 grid. QualNet was modified to reflect the two-disc interference model with a reception range of 250 m and interference range of 550 m. It can be seen that the decomposed formulation provides high quality routes when compared to the standard routes and the performance is comparable to the basic formulation. Similar results were also observed in end-to-end delay and jitter of the connections.

We now compare the performance of the decomposed formulation in larger networks where it is infeasible to run the basic model. Fifty random networks, each with 200 nodes in a 1500 m$\times$1500 m network with 10 random CBR connections of approximately 3-4 hops were evaluated for varying traffic rates.

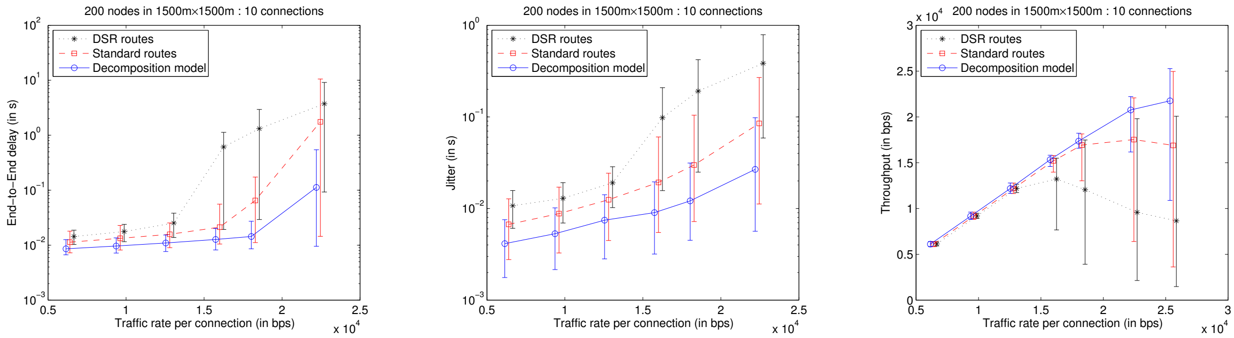
Figure 2 summarizes the significant gains in the end-to-end delay, jitter and throughput of the routing configuration obtained from the decomposition formulation when compared with the standard routes and DSR routing protocol. Delay and jitter metrics are substantially lower in both unsaturated and saturated traffic conditions. As expected, the formulation results in greater throughput under higher degree of saturation of the network. The end-to-end delay of the proposed formulation remains stable for higher traffic

[1]Standard routes are the most frequently observed routes on DSR protocol [14] in QualNet simulator. Such routes were converted into static routes to eliminate the overhead of the routing protocol and provide a fairer comparison.



(a) Run-time study (b) Throughput study

Fig. 1. Comparing Basic Model and Decomposition: 3 *orders* of improvement in the runtime of decomposition model. The performance of the decomposed routes are comparable to *basic model* and outperform *standard routes*



(a) End-to-end delay (b) Jitter (c) Throughput

Fig. 2. Performance evaluation in larger random networks: In an unsaturated network, decomposition model has 25% lesser end-to-end delay and 40% lesser jitter in comparison with standard routes. Delay is 15 times lesser, jitter is 3 times lesser and throughput is 30% more in saturated network.

while there is an exponential increase in the standard routes indicating that the routes obtained from the above formulation can support greater traffic demands.

The evaluation of the formulation under a realistic protocol stack and differing traffic conditions indicates that the greedy approach of routing is suboptimal and the non-greedy algorithms that detect neighboring interference levels, by metrics such as busy-time, can be employed to improve routing in MHWNs.

VI. Discussion and Future Work

Few approximations and assumptions were made in the formulation in order to enable a feasible and practical solution to the single single-path routing problem in MHWNs. In this section, we discuss their implications on the global optimal solution and sketch our future plans to extend the model.

Exact modeling of interference is known to be NP-hard even when the active links in the route are known and under simple interference models like unit-disc model of interference [2], [4]; it is an instance of the *Maximal Independent Sets* problem in graph theory. Hence, precise estimation of the effect of interference in a routing problem where active links needs to be solved is infeasible. In order to overcome this complexity, we approximate the effect of interference by busy-times observed at a link. Since busy-times are theoretically hard to formulate but relatively easy to measure, the measured values can be substituted in the routing framework to enable optimal distributed protocols even when the theoretical formulation is NP-hard. In future work, we like to evaluate a distributed routing protocol based on the above framework, but with measured busy-time values.

Wired networks have widely used the Kleinrock delay equation [15] to capture the queuing delay. The problem with direct application of such an equation is the assumption that the queuing delay is only a function of the capacity of the link and the flow on the link. In MHWNs this is not true, since the delay is also a function of the traffic on the interfering links, which makes the objective function non-convex. Formulation and evaluation of the model by approximating the queuing delay is a part of our future work.

VII. Concluding Remarks

Interference aware network flow models are promising for analysis and optimization of MHWNs. However, the computational complexity of these models prevents their use in realistic large scale scenarios. In this paper, we formulated a practical routing model that can be solved in polynomial time using a decomposition based approach. Primal decomposition was employed where the sub-problems abstract the connection identification rules of the routing protocol and the master problem mimics the routing metric update rules in the network. The resulting formulation was shown to be several orders of magnitude more efficient than the basic formulation. The quality of the routes obtained from the formulation was comparable to the basic MCF formulation and outperforms the commonly used routing configurations under saturated and unsaturated traffic conditions.

We believe that the formulation serves as a scalable optimization framework for analysis and QoS-based routing in practical MHWNs. The derived rules of the routing protocol should aid in the design of optimal protocols.

Acknowledgements

This work was in part financially supported by European Union (ARAGORN project). We acknowledge also the partial support from DFG and RWTH Aachen through UMIC-research center facility.

References

[1] M. Chiang, S. H. Low, A. R. Calderbank, and J. C. Doyle, "Layering as optimization decomposition: A mathematical theory of network architectures," in *Proceedings of IEEE*, 2007.

[2] K. Jain, J. Padhye, V. N. Padmanabhan, and L. Qiu, "Impact of interference on multi-hop wireless network performance," in *MobiCom*. New York, NY, USA: ACM Press, 2003, pp. 66–80.

[3] M. Kodialam and T. Nandagopal, "Characterizing achievable rates in multi-hop wireless networks: the joint routing and scheduling problem," in *MobiCom*. New York, NY, USA: ACM Press, 2003, pp. 42–54.

[4] L. Chen, S. H. Low, M. Chiang, and J. C. Doyle, "Cross-layer congestion control, routing and scheduling design in ad hoc wireless networks," in *INFOCOM*, 2006.

[5] V. Kolar and N. B. Abu-Ghazaleh, "A Multi-Commodity Flow Approach for Globally Aware Routing in Multi-Hop Wireless Networks," in *PERCOM*. Washington, DC, USA: IEEE Computer Society, 2006, pp. 308–317.

[6] ——, "Towards interference-aware routing for real-time traffic in multi-hop wireless networks," in *DS-RT*. Washington, DC, USA: IEEE Computer Society, 2007, pp. 80–83.

[7] A. Schrijver, *Theory of linear and integer programming*. New York, NY, USA: John Wiley & Sons, Inc., 1986.

[8] D. S. J. D. Couto, D. Aguayo, J. Bicket, and R. Morris, "A high-throughput path metric for multi-hop wireless routing," in *MobiCom*. New York, NY, USA: ACM Press, 2003, pp. 134–146.

[9] L. Chen and W. Heinzelman, "QoS-aware routing based on bandwidth estimation for mobile ad hoc networks," *Selected Areas in Communications, IEEE Journal on*, vol. 23, no. 3, pp. 561–572, March 2005.

[10] R. K. Ahuja, T. L. Magnanti, and J. B. Orlin, *Network flows: theory, algorithms, and applications*. Prentice-Hall, Inc., 1993.

[11] A. Nasipuri, R. Castaneda, and S. R. Das, "Performance of multipath routing for on-demand protocols in mobile ad hoc networks," *Mob. Netw. Appl.*, vol. 6, no. 4, pp. 339–349, 2001.

[12] D. Bertsekas and R. Gallager, *Data networks*. Prentice-Hall, Inc. Upper Saddle River, NJ, USA, 1987, pp. 379–381.

[13] "Qualnet network simulator, version 3.8," http://www.scalable-networks.com/.

[14] D. B. Johnson, D. A. Maltz, Y.-C. Hu, and J. G. Jetcheva, "The Dynamic Source Routing Protocol for Mobile Ad Hoc Networks (DSR)," 2002.

[15] L. Kleinrock, *Communication nets; stochastic message flow and delay*. Dover Publications, Incorporated, 1972.

Appendix
Decomposing the objective function

Let the original objective function (Equation 5) be denoted by $f(\boldsymbol{X}, \mathcal{C}, \mathcal{E}, \mathcal{B}, \mathcal{W})$ where $\boldsymbol{X}$ is the matrix of variables x^k_{ij}.

$$\begin{aligned} f = & \sum_{m\in\mathcal{C},(i,j)\in\mathcal{E}} p_m A^m_{ij} \\ = & \sum_{m\in\mathcal{C},(i,j)\in\mathcal{E}} p_m 1^m_{ij}\big(\sum_{k\in C} x^k_{ij} + \sum_{(y,z)\in\mathcal{W}_{ij},k\in\mathcal{C}} x^k_{yz}\big) \\ = & \sum_{m\in\mathcal{C}} p_m \Big(\big(\sum_{(i,j)\in\mathcal{E}} 1^m_{ij} \sum_{k\in C} x^k_{ij}\big) \\ & + \big(\sum_{(i,j)\in\mathcal{E}} 1^m_{ij} \sum_{(y,z)\in\mathcal{W}_{ij},k\in\mathcal{C}} x^k_{yz}\big)\Big). \end{aligned} \quad (10)$$

Let $g(m)$ denote the second part of Equation 10

$$g(m) = \sum_{(i,j)\in\mathcal{E}} 1^m_{ij} \sum_{(y,z)\in\mathcal{W}_{ij},k\in\mathcal{C}} x^k_{yz}$$

and renaming (i,j) to (a,b) yields

$$= \sum_{(a,b)\in\mathcal{E}} \sum_{(y,z)\in\mathcal{W}_{ab}} 1^m_{ab} \sum_{k\in\mathcal{C}} x^k_{yz}.$$

Let 2^{ab}_{yz} be an indicator function that is 1 if $(y,z) \in \mathcal{W}_{ab}$. Renaming (y,z) to (i,j) and interchanging the summation terms, we get

$$= \sum_{(i,j)\in\mathcal{E}} \sum_{(a,b)\in\mathcal{E}} 2^{ab}_{ij} 1^m_{ab} \sum_{k\in\mathcal{C}} x^k_{ij}.$$

The term $\sum_{(a,b)\in\mathcal{E}} 2^{ab}_{ij}(\ldots)$ are all the edges (a,b) that interfere with some edge (i,j). Hence, we have

$$\begin{aligned} & = \sum_{(i,j)\in\mathcal{E}} \sum_{k\in\mathcal{C}} x^k_{ij} \big(\sum_{(a,b)\in\mathcal{B}_{ij}} 1^m_{ab}\big) \\ g(m) & = \sum_{(i,j)\in\mathcal{E}} n^m_{ij} \big(\sum_{k\in\mathcal{C}} x^k_{ij}\big), \end{aligned} \quad (11)$$

where n^m_{ij} is the number of *active* links in connection m that interfere with (i,j). Substituting the value of $g(m)$ from Equation 11 in the original objective function Equation 10, we have:

$$\begin{aligned} f = & \sum_{m\in\mathcal{C}} p_m \Big(\big(\sum_{(i,j)\in\mathcal{E}} 1^m_{ij} \big(\sum_{k\in C} x^k_{ij}\big)\big) \\ & + \big(\sum_{(i,j)\in\mathcal{E}} n^m_{ij} \big(\sum_{k\in\mathcal{C}} x^k_{ij}\big)\big)\Big) \\ = & \sum_{m\in\mathcal{C}} p_m \sum_{(i,j)\in\mathcal{E}} \big(1^m_{ij} + n^m_{ij}\big)\big(\sum_{k\in C} x^k_{ij}\big). \end{aligned}$$

Interchanging summation terms and rearranging, we have:

$$\begin{aligned} = & \sum_{k\in C} \sum_{(i,j)\in\mathcal{E}} \Big(\big(\sum_{m\in\mathcal{C},m\neq k} p_m (1^m_{ij} + n^m_{ij})\big) \\ & + \big(p_k(1^k_{ij} + n^k_{ij})\big)\Big) x^k_{ij} \\ f = & \sum_{k\in C} \sum_{(i,j)\in\mathcal{E}} (c^{k^*}_{ij} + s^{k^*}_{ij}) x^k_{ij}, \end{aligned} \quad (12)$$

where $c^{k^*}_{ij} = \sum_{m\in\mathcal{C},m\neq k} p_m(1^m_{ij} + n^m_{ij})$ is the weight for the flow on the edge (i,j) that is introduced due to cross-connection interference. Similarly, the self-connection interference weight is given by $s^{k^*}_{ij} = p_k(1^k_{ij} + n^k_{ij})$.