ALL OPTIMAL SOLUTIONS IN STDMA SCHEDULING

Diogo Quintas, Vasilis Friderikos

Centre for Telecommunications Research,
King's College London, Strand, WC2R 2LS, London, England.

ABSTRACT

In this paper we review different Mixed Integer Programming formulations of the STDMA Scheduling problem and introduce a novel formulation. It is shown that the problem admits, in general, multiple optimal solutions - we propose an efficient cut generation procedure to construct all optimal schedules and investigate the properties of optimal schedules in two small networks.

Index Terms— mixed integer programming, scheduling, all optima, physical interference

1. INTRODUCTION

Scheduling is a fundamental problem in wireless networks. Due to the broadcasting nature of the wireless medium the performance of the entire network is affected by the scheduling algorithms that are used and it is not surprising that researchers have devoted a lot of attention to this problem. Nonetheless, fundamental questions are still open research questions - in particular what are the characteristics of optimal schedules. Mixed Integer Programming models are unsuitable to be used in real time, nonetheless these models can give insight to the structural properties of the problem and serve as benchmarks in assessing (non optimal) low complexity algorithms.

In this paper we concentrate on the minimum frame length scheduling problem and present a new formulation that leads to more stable models. The models are used to develop an algorithm to generate all optimal solutions to the problem that can then be used in investigating the properties of optimal schedules.

2. PROBLEM DESCRIPTION

The concept of the Spatial-TDMA scheduling problem has been formalized in [1]. In the classical form of the problem the objective is to utilize as little number of time slots as possible, and thus minimizing delay and maximizing throughput. Equivalent forms of the problem consist in finding the maximum number of transmissions in a given number of slots (in particular with one slot), or maximizing a weighted sum of the number of transmissions (taking into account different priority levels). In this paper we focus on the classical version: the minimum frame length scheduling problem (MFLSP). Previous work in this problem can be broadly classified according to the model for interference used.

The most common interference model in the MFLSP literature is the range or disk model [2,3]. An interference range for each of the transmitters is defined and no receiver maybe active if it is within the range of a active transmitter. A conflict graph can be constructed where for each wireless link there is a vertex and edges connect two pairs that cannot be active at the same time; then graph theoretical techniques can be used to solve the problem. The advantages of these models are clear. First a long body of graph theoretical literature exists allowing for the development of low complexity scheduling protocols. Secondly, by abstracting the interference mechanism these models can be deemed flexible in modeling a variety of situations. It is not without disadvantages: setting the range is not straightforward and interference is seen as a binary (and a local phenomenon) and can not take into account accumulated (aggregated) interference. It has been shown that the performance of protocols based in the graph model may lead to infeasible schedules or underutilized transmission slots dropping in that sense the performance of the network [4].

A more realistic interference model is the physical interference model [5–7]. In this model interference is modeled explicitly via the signal to interference plus noise ratio (SINR). Given a power level p_l and effective link gains g_{kl} between the transmitter of link k and the receiver of link l the SINR is given by:

$$\frac{g_{ll}p_l}{W+\sum_{k\neq l} g_{kl}p_k} \tag{1}$$

If the above expression, where W denotes the lump sum noise level at the receivers, is above a threshold γ for all active links then communication is feasible. This model has the flexibility of allowing different power level strategies.

The work reported in this paper has formed part of the Green Radio Core 5 Research Programme of the Virtual Centre of Excellence in Mobile & Personal Communications, Mobile VCE, www.mobilevce.com . This research has been funded by EPSRC and by the Industrial Companies who are Members of Mobile VCE.

2.1. Complexity Status

The broadcast scheduling problem under the various models has been shown to be hard. In graph based interference models the MFLSP is simply a graph coloring problem - assigning slots to each link corresponds to coloring the conflict graph induced by the interference model. The graph coloring problem is one of Karp's original NP-Complete problems [8]. In the SINR model the complexity status depends on how the gain factors are modeled. For arbitrary gain factors the authors in [7] have shown that the graph coloring problem can be reduced to the MFLSP for a fixed power allocation, nonetheless the reduction the authors propose can be adapted to allow power control. Hence the MFLSP for arbitrary gain factors is in NP-Complete. Where the gain factors need to respect a triangular inequality (Geometric SINR model), the MFLSP with a uniform power allocation has been shown to be in NP-Complete [5]. The complexity status for the MFLSP with a variable power allocation in the Geometric SINR model is still an open question.

3. MIP FORMULATIONS

The minimum frame scheduling problem with joint power allocation lends itself to a MIP formulations due to the binary decision of which slot to schedule each link and the decision on the power levels of each transmitting pair.

3.1. Standard Formulation

Let y_t with $t = 1, ...|L|$, where L is the set of links to be scheduled, be a 0-1 variable indicating if slot t is activated or not. Furthermore, let x_{tl} be a binary variable indicating if link l is activated in slot t, and p_{tl} is the power level at which the link transmits. The MFLSP can be formulated as:

$$\text{Minimize,} \sum_t y_t \tag{2}$$

$$\text{Subject to, } p_{tl} \leq P_{max} x_{tl} \qquad \forall l \in L, t = 1, .., |L| \tag{3}$$

$$p_{tl} \geq \frac{W\gamma}{g_{ll}} x_{tl} \qquad \forall l \in L, t = 1, .., |L| \tag{4}$$

$$\frac{g_{ll} p_{tl} + (1 - x_{tl})M}{W + \sum_{k \neq l} g_{kl} p_{tl}} \geq \gamma \qquad \forall l \in L, t = 1, .., |L| \tag{5}$$

$$\sum_t x_{tl} \geq 1 \qquad \forall l \in L \tag{6}$$

$$x_{tl} \leq y_t \qquad \forall l \in L, t = 1, .., |L| \tag{7}$$

$$p_{tl} \geq 0, x_{tl} \in 0, 1, y_t \in 0, 1 \qquad \forall l \in L, t = 1, .., |L| \tag{8}$$

Constraints (3) and (4) are binding constraints for the variables x_{tl} and p_{tl}, constraint 5, where M is a big enough constant, ensures all active links in any time slot satisfy the signal to noise ratio threshold, constraint (6) requires that all links are active at least during one slot, i.e. that all transmission requests are satisfied in the frame. Constraint (7) guarantees that if a link is active in any given slot then the corresponding slot is activated. Minimizing the sum of active slots ensures that the optimal solution is the minimum frame length schedule.

3.2. Primal Decomposition Formulation

The MFLSP can be seen as a minimum set cover problem. Note that each schedule can be seen as a set of link activation vectors in each slot: a set of sub sets of (feasible) active links, called transmission groups. Hence, the minimum frame length schedule is simply the minimum of such sets where all links transmit at least once:

$$\text{Minimize,} \sum_g \lambda_g \tag{9}$$

$$\text{Subject to,} \sum_g a_{gl} \lambda_g \geq 1 \qquad \forall l \in L \tag{10}$$

$$\lambda \in 0, 1 \tag{11}$$

In the above formulation λ_g is a binary variable indicating if transmission group is in the schedule or not, while a_{gl} is a parameter taking the value 1 if link l is active in group g and 0 otherwise. This type of formulation requires all transmission groups to be identified - note that the number of transmission groups grows exponentially with the number of links. The authors in [7] proposed a column generation algorithm to solve the MFLSP in this formulation that does not require to generate all transmission groups. In each step of their column generation algorithm a transmission group is generated by solving the original model for one slot only and with a weighted sum of all active links for an objective function - the weights being given by the dual values of the restricted set cover formulation.

3.3. Numerical Instability Issues

The SINR constraint (5) creates difficulties for these models. First, the constant M is hard to set at an appropriate value - if set to a high value it leads to poor LP relaxations, if set to a too low value it makes the problem infeasible. Second, the gain factors are (usually) small numbers of the order of 10^{-10} - this leads to a numerical unstable model.

An alternative model for the SINR constraint is to check necessary and sufficient feasibility conditions for a transmission group that are given by the power control literature [9]. Instead of modeling the SINR criterion directly, infeasible transmission groups are excluded by generating 0-1 constraints; these constraints eliminate all the infeasible transmission groups. All minimal infeasible transmission groups[1] g need to be identified and the constraint $\sum_{l \in g} x_l \leq |g|$ is

[1] A minimal infeasible transmission group is a transmission group such that if any of the links is removed the transmission group becomes feasible

added for each. This model, leads to an exponential number of constraints and a constraint generation procedure needs to be used to solve the model. The infeasible groups of cardinality two and three can easily be identified and the model is solved; once an integer solution is obtained the feasibility of each identified group is checked using the criterion above. Then, for each infeasible group the minimal subset(s) of infeasible links in that group are identified and the corresponding cuts are added - the algorithm stops when all transmission groups in the constructed schedule are feasible.

A similar idea has been pursued in [10]. In that work, the authors discretized the power variable and re-formulated the problem with an exponential number of constraints. The advantage of their model is that power levels are computed explicitly, however extra integer variables are added further complicating the model. In the procedure we propose no extra integer variable is needed and the model is simplified by dropping the power variables. Note that computing the necessary power levels from a given schedule is a straightforward matter.

4. GENERATING ALL OPTIMAL SCHEDULES

Because it is possible to have an infinite number of optimal solutions with distinct real part, we consider that two optimal points are only different if and only if they have distinct integer parts in MIP models. Let z be a binary vector and z^s represent a optimal solution $s \in S$, where S is the set of all optimal solutions that have been identified. A sequential algorithm to find all optima has been proposed ([11] and references therein). The sequential algorithm proceeds to solve the model in each step excluding all the previous found solutions z^s. The algorithm stops when no more solutions can be found. To exclude the point in the unity hypercube representing the solution the following constraint is added,

$$\sum_{i:z_i^s=0} z_i + \sum_{i:z_i^s=1} (1-z_i) \geq 1 \forall s \tag{12}$$

The first summation term is greater or equal to 1 if for some variable $z_i^s = 0$ we have, in the current solution, $z_i = 1$ and, similarly, the second summation term is greater or equal to 1 if for some variable $z_i^s = 1$ we have, in the current solution, $z_i = 0$ - thus the constraint guarantees that at least one of the variables takes a different value than in any of the previously identified solutions. The main strength of this algorithm is the fact that it allows for a different objective function and thus it allows for solutions with desirable properties to be generated. Note however that the proposed cut only excludes a single point in the unit hypercube - when certain permutations of values in the binary vector lead to 'equivalent' solutions the cut is not sufficient to generate distinct solutions, this is the case in the MFLSP.

4.1. Generating distinct minimum length schedules

In the MFLSP any feasible point represents a family of equivalent schedules given by all possible permutations of the transmission groups that the solution identifies. Therefore the added constraint(s) should not only exclude the identified optimal point but all permutations. Given that the optimal frame length schedule is entirely composed by distinct transmission groups, using the cut given by Ineq. (12) for each permutation, we would add $T^*!$ constraints, where T^* is the minimum frame length, for each identified optimum. To exclude all possible permutations we have to ensure that at least one of the transmission group g given by the optimal solution s is not present in any of the slots. Let θ_g^s be a binary vector with $|L|$ components taking the value 1 if link l is active in transmission group g in solution s and 0 otherwise. Let y_g be a binary variable taking the value 1 if group g is not in the schedule and 0 otherwise and let $\mathcal{F}_s$ be the set of groups in solution $s \in S$. The following constraints guarantee at least one group is not reused:

$$\sum_{l:\theta_{lg}^s=0} x_{tl} + \sum_{l:\theta_{lg}^s=1} (1-x_{tl}) \geq y_g \forall t,g,s \tag{13}$$

$$\sum_{g\in\mathcal{F}_s} y_g \geq 1 \forall s \tag{14}$$

Constraint (13) guarantees that if group g is used in any of the slots y_g has to be 0 and constraint (14) ensures that at least one of the groups is not being used. For each identified optimum we add T^*N+1 new constraints and N new variables, where N is the number of newly identified transmission groups.

Ideally the new generated solutions should be as distinct as the one that was previously found. Traditionally this is done by maximizing the hamming distance between the previous solutions and the newly generated solution. In the case of the wireless scheduling problem this does not necessarily maximize the difference between two schedules; intuitively the difference between two schedules is given by the number of distinct transmission groups in each. Therefore if we were to maximize the number of previously identified transmission groups not used in the new solution, we can guarantee that the solution is as original as it can be. The number of transmission groups that are not in use in the new solution is simply given by the sum of all variable y_g: $\sum_g y_g$. A further benefit of this approach is that it gives a simple way of identifying when all groups that can be part of an optimal solution have been generated - this can lead to a faster procedure to generate the remaining optimal schedules.

4.2. A Sequential Algorithm based on Primal Decomposition

A faster algorithm can be devised by noting that after having identified all transmission groups that are part of optimal schedules, subsequent solutions are just alternative feasible combinations of the identified groups. Different combinations of the transmission groups can then be generated, excluding the previously found ones. This achieves a dimension reduction of $|L|T^*$ in the size of the problem. We can think of this procedure as a two phases procedure: in the first phase we are computing different schedules with different groups and in the second phase we are computing all the feasible combinations of transmission groups.

For the first phase we proceed as in the previous case, by solving the same MIP with added constraints (13) and instead of excluding a particular combination of groups as with constraint (14), we guarantee that at least one of the groups is not in use, ensuring that a newly proposed solution has identified a new group. We let $\mathcal{G}$ be the set of all identified groups and require that at least one element is not active: $\sum_{g\in\mathcal{G}} y_g \geq 1$.

The first phase of the algorithm stops when no more feasible solutions can be found, i.e. when the model becomes infeasible. The infeasibility ensures that all the solutions with distinct groups have been found and unfound solutions are thus just alternative feasible combinations. To generate an alternative combination we use a set cover like formulation as in section 3.2 and add the constraint $\sum_{g\in\mathcal{F}_s} \lambda_g \geq 1 \forall s$. The second phase continues until $\sum_g \lambda_g^* > T^*$, i.e. there are no more schedules of length T^*.

5. ILLUSTRATIVE EXAMPLES

Two small networks are investigated, where nodes have been placed randomly in a 1km by 1km area and schedules are constructed where each link is active at least once. Connectivity is created by triangulating all nodes, thus creating a fully connected network. The effective link gains are modelled using a distance based path loss model with a path loss exponent of 3.5 (appropriate to model propagation conditions in urban environments).

The first network has 7 nodes and 11 links (due to space limitation we only show the topology of the bigger network). Eleven minimum frame length schedules with 8 time slots where each link transmits at least once are identified, of which 5 schedules have more than one transmission per link (hence offering higher throughput). Looking at the minimum power consumption that each schedule can offer, in the schedules where each link transmits exactly once, the power transmitted by the most expensive schedule was more than twice as much of the least expensive one, while for the schedules with 12 scheduled transmissions the least expensive schedule has 27% less power consumption than the most expensive one.

The second network we investigate has 10 nodes and 21 links (see figure 1). Even though the increase in the size is not that significant we can now identify more than 1000 solutions with 12 time slots each. Note that only 91 distinct transmission groups were identified (see figure 2) and distinct schedules are just different combinations of the identified groups.

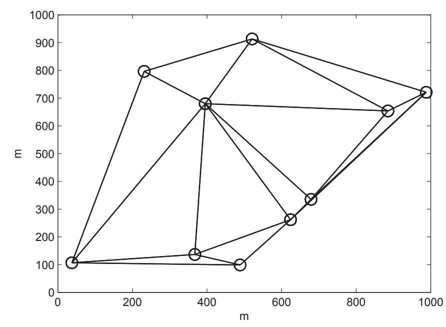

Fig. 1: Topology and location of nodes in the example network with 10 nodes and 21 links.

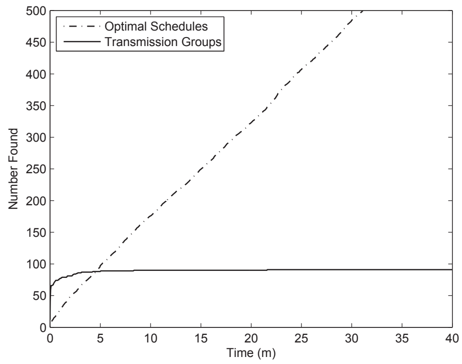

Fig. 2: The runtime to generate 1000 solutions. All transmission groups are generated in the first 5 minutes - from then on all identified schedules are different combinations of the identified groups.

In this network a wider variance between the quality of each schedule for different criteria is observed - herein we analyze three: robustness, throughput and power consumption.

We have found schedules with up to 35 transmissions scheduled - 66% more network throughput than in the minimum required. As in the previous example, remarkable differences exist between the total power consumption of the network in each schedule. In figure 3 we show box plots of the power consumption of each schedule categorized by the number of transmissions in the schedule. As expected, the median power consumption grows with the number of transmissions performed. Nonetheless, big differences within each throughput level are observed - with an overlapping range across throughput levels. Across all throughput levels the ratio between the most power consuming schedule and the least power consuming is 2.5. To assess the robustness of each schedule to fading, we computed the probability of no outages occurring due to Rayleigh fading, i.e. the probability of not having a feasible power allocation given that the effective link gains might suffer from Rayleigh fading.

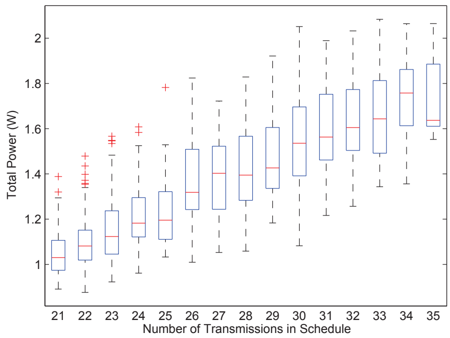

Fig. 3: Box plots for the minimum transmitted power in each generated schedule for each number of transmissions in the schedule.

We find that in the most robust schedule an outage is 149.4 times less likely to occur than in the least robust schedule - the most robust schedule has 21 transmissions and the least robust 31. Naturally the more transmissions occur, the less robust a schedule will be, due to over packing at each slot. Nonetheless even within each number of transmissions a great variability is found. For schedules with 23 transmissions the ratio between the probability of no outages of the most and least robust schedules is 20.4 (the biggest difference) and, for schedules with 35 transmissions the ratio is 2.8 (the lowest difference).

6. FINAL REMARKS

In this paper Mixed Integer Programming formulations of the MFSLP were briefly reviewed. We have identified a new model that is more numerically stable (the constraint matrix coefficients are just 0 or 1), with the penalty of having an exponential number of constraints - suggesting a constraint generation procedure to solve the model. In order to study the properties of optimal schedules we detailed a cut generation procedure to construct all optimal solutions.

The problem admits, in general, multiple optimal solutions. The existence of multiple optimal solutions has practical implications to system and protocol design. In our small examples we have shown that optimal solutions exist to the MFLSP with a wide difference in their power consumption, throughput and robustness. This indicates that there is a great scope to improve the performance of existing scheduling algorithms by ensuring that the constructed schedules are Pareto optimal. A particularly interesting Pareto solution, in the context of energy efficiency, is the minimum frame schedule at the least power consumption. The algorithms we proposed to generate all optimal solutions can be used in studying the production possibility frontier of wireless networks and can be an important tool in the design and planning of wireless networks. They are also useful in providing benchmark solutions to multi objective optimization algorithms. As future avenues of research, heuristics need to be developed that are able to explore the entire Pareto frontier and are able to trade-off the conflicting objectives.

7. REFERENCES

[1] R. Nelson and L. Kleinrock, “Spatial TDMA: A collision-free multihop channel access protocol,” *Commun., Trans.*, vol. 33, no. 9, pp. 934 – 944, sep 1985.

[2] A. Ephremides and T. Truong, “Scheduling broadcasts in multihop radio networks,” *Commun,, Trans.*, vol. 38, no. 4, pp. 456 –460, apr 1990.

[3] S. Sarkar and S. Ray, “Arbitrary throughput versus complexity tradeoffs in wireless networks using graph partitioning,” *Aut. Control, Trans.*, vol. 53, no. 10, pp. 2307–2323, Nov. 2008.

[4] J. Grönkvist and A. Hansson, “Comparison between graph-based and interference-based STDMA scheduling,” in *MobiHoc*. New York, NY, USA: ACM, 2001, pp. 255–258.

[5] O. Goussevskaia, Y. A. Oswald, and R. Wattenhofer, “Complexity in geometric SINR,” in *MobiHoc*. NY, USA: ACM, 2007, pp. 100–109.

[6] G. Brar, D. M. Blough, and P. Santi, “Computationally efficient scheduling with the physical interference model for throughput improvement in wireless mesh networks,” in *MobiCom*. New York, NY, USA: ACM, 2006, pp. 2–13.

[7] P. Björklund, P. Värbrand, and D. Yuan, “A column generation method for spatial TDMA scheduling in ad hoc networks,” *Ad Hoc Net.*, vol. 2, no. 4, pp. 405 – 418, 2004.

[8] R. Karp, “Reducibility among combinatorial problems,” in *Complexity of Computer Computations*, R. Miller and J. Thatcher, Eds. Plenum Press, 1972, pp. 85–103.

[9] J. Zander, “Performance of optimum transmitter power control in cellular radio systems,” *Veh. Tec., Trans.*, vol. 41, no. 1, pp. 57 –62, feb 1992.

[10] F. DAndreagiovanni, “Pure 0-1 programming approaches to wireless network design,” *4OR: Quart. Jour. of Op. Res.*, pp. 1–2, doi: 10.1007/s10288-011-0162-z. [Online].

[11] E. Danna, M. Fenelon, Z. Gu, and R. Wunderling, “Generating multiple solutions for mixed integer programming problems,” in *Integer Programming and Combinatorial Optimization*, 2007, pp. 280–294.