Cross-Layer Design for Wireless Mesh Networks Using Column Generation

Christelle Molle, Fabrice Peix, Hervé Rivano

INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

Cross-Layer Design for Wireless Mesh Networks Using Column Generation

Christelle Molle — Fabrice Peix — Hervé Rivano

I3S(CNRS-UNSA)/INRIA MASCOTTE project, Sophia Antipolis, FRANCE.

E-mail: {christelle.molle, fabrice.peix, herve.rivano}@sophia.inria.fr.

N° 6448

Décembre 2007

Thème COM

Rapport de recherche

ISSN 0249-6399 ISRN INRIA/RR--6448--FR+ENG

Cross-Layer Design for Wireless Mesh Networks Using Column Generation*

Christelle Molle† , Fabrice Peix , Hervé Rivano
I3S(CNRS-UNSA)/INRIA MASCOTTE project, Sophia Antipolis, FRANCE.
E-mail: {christelle.molle, fabrice.peix, herve.rivano}@sophia.inria.fr.

Thème COM — Systèmes communicants
Équipe-Projet Mascotte

Rapport de recherche n° 6448 — Décembre 2007 — 16 pages

Abstract: Wireless Mesh Networks (WMNs) have become an interesting answer for broadband wireless networking. Cross-layer optimization problems for WMNs deployment and management are necessary and challenging. In this paper we focus on jointly optimizing routing and link scheduling in a single-channel wireless mesh network, in order to maximize fair network throughput or equivalently minimize time period. Our approach is based on a *path/configuration linear formulation* of the joint routing and scheduling problem, which is solved by column generation with two auxiliary programs to generate new paths and configurations. The method is validated on small topologies from an optimal *node/arc formulation*, and simulations are then done on random and grid topologies.

Key-words: Wireless Mesh Networks, Routing, Scheduling, Integer Linear Programming, Column Generation.

* This work has been partially funded by european project IST/FET AEOLUS, ANR-JC OSERA, and ARC CARMA.
† C. Molle PhD is funded by DGA, France.

Centre de recherche INRIA Sophia Antipolis – Méditerranée
2004, route des Lucioles, BP 93, 06902 Sophia Antipolis Cedex (France)
Téléphone : +33 4 92 38 77 77 — Télécopie : +33 4 92 38 77 65

Conception Cross-Layer de Réseaux Radio Maillés par Génération de Colonnes

Résumé : Dans cet article, nous étudions la capacité des réseaux radio maillés. Nous présentons un modèle linéaire optimal du problème du routage et de l'ordonnancement des communications dans un réseau radio maillé synchrone à un seul canal radio. L'objectif est de minimiser la taille de la période de temps considérée de manière à satisfaire des demandes de clients souhaitant accéder à Internet. Nous présentons une formulation relachée *chemin/stable* du problème combinant le routage et l'ordonnancement des communications afin d'éviter les collisions qui peuvent apparaitre en technologie radio. Etant donnée la taille exponentielle des variables, la résolution du problème utilise la génération de colonnes qui permet de partir d'un sous-ensemble de variables et d'ajouter à chaque itération un nouveau chemin et/ou un nouveau stable améliorant l'objectif. La méthode est validée par comparaison à une formulation optimale en sommet/arc sur des topologies de petite taille. Les simulations sont ensuite effectuées sur des topologies en grille ou aléatoires.

Mots-clés : Réseaux Radio Maillés, Capacité, Routage, Ordonnancement, Programmation Linéaire, Génération de Colonnes.

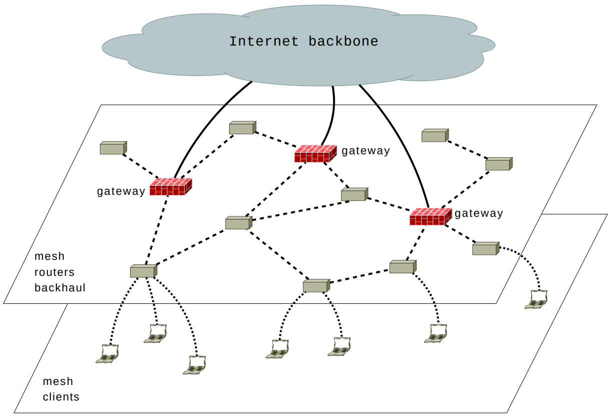

Figure 1: Exemple of a WMN topology

1 Introduction

Wireless mesh networks (WMNs) are cost-effective solutions for ubiquitous high-speed services [1]. They are self-organized networks with a fixed infrastructure of wireless mesh routers (MRs) interconnected to provide Internet access to mobile network users. This infrastructure, forming a wireless backhaul network, is integrated with Internet by special routers called mesh gateways (MGs). Thus mesh clients (MCs) access Internet by multi-hop communications through the backhaul. A WMN can be represented as a two-tier architecture as shown in Figure 1, and a logical separation is maintained between MCs to MRs one-hop connections and MRs to MGs multi-hop communications. This work focuses on WMN backhaul optimization because of its stationary characteristic and clients demands are modelled by some weight on each MR node. We globally call mesh points (MPs) the union of MRs and MGs.

There is an intensive research in providing full optimization frameworks for WMNs. Network throughput is one of the most important issues when evaluating the performance of a WMN. But one of the major problem facing wireless networks is the capacity reduction due to wireless contentions [7, 10]. In [6], Gupta and Kumar use an analytical approach to prove that the per node capacity of a random wireless ad-hoc network decreases as $\mathcal{O}(\frac{1}{\sqrt{n}})$ as the size of the network, n, grows. In WMNs, capacity bottlenecks appear around the gateways, therefore the available capacity for each node is reduced to $\mathcal{O}(\frac{1}{n})$ [8].

In addition, network performances strongly depends on cross-layer optimization including deployment strategies, routing protocols, and physical properties. Capacity evaluation is useful to improve the performance of routing protocols, and routing efficiency depends on the problem on allocating physical and data link layer resources. Interferences between link transmissions clearly impact the routing used to satisfy traffic demands. In the same way traffic routing determines the traffic flows for each link, but it has to be done in a way such that the communication requirements for the links can be met.

In order to maximize network capacity, the optimal allocation of resources has therefore to be determined. In a spatial Time-Division-Multiple-Access scheme (STDMA), the transmission capacity is divided into time slots, and each link is assigned some dedicated slots. In this so-called *link scheduling problem*, one has to generate a feasible transmission schedule to route a given arbitrary network traffic distribution at maximum rate [13]. At each slot, only a set of pairwise non-interfering links is selected, leading to a schedule that dictates when links transmit. These sets are called *configurations* and the set of all possible configurations is exponential. Finding an optimal schedule is thus intractable for network with more than 15 links and optimization-based approaches trying to reduce the size of the space of configurations have been proposed [16] in conjunction with approximation algorithms [9].

Column generation is a prominent technique to cope with large-scale integer programs [14]. When the number of variables is exponential, it provides a decomposition of the problem into master and sub-problems. The master is a different formulation of the original problem depending on combinatorial structures that can be generated through the sub-problems, also called auxiliary problems. The continuous relaxation of the master problem is solved in order to obtain dual variables and corresponding induced costs. Then, given the set of dual variables, either identify a column that has a favorable induced cost, or indicate that no such column exists.

The joint routing and scheduling problem (JRSP) has been extensively studied in the litterature and column generation has already been applied to solve the continuous relaxation of this problem [17, 2]. However, the goal was to minimize time period length given a fixed number of packets to send between the nodes, and the packets destination was fixed for every sender. In a WMN with multiple gateways, MRs usually send its aggregated traffic to different gateways through multi-paths [12, 15].

In this work, the formulation can either minimize time period length, or maximize the maximum fair throughput at each source node, saying the maximum number of packets that can be routed from each MR to MGs. Generating not only new configurations, but also new paths in our column generation process, allows to choose longer paths that can carry more flows. This path/configuration formulation has no longer been proposed, and the technique presented here constitute a method to compute optimal solutions for the routing and scheduling problem.

2 Problem Definition and Assumptions

The network can be viewed as a directed graph $G = (V, E)$ of N nodes representing mesh points. Each device has a single interface, sharing a single channel with other nodes. Let $V = V_r \cup V_g$ where V_r denotes the set of mesh routers, and V_g the set of mesh gateways. For each transmission, a MR from V_r will send data packets to MGs from V_g through multi-hop paths.

We suppose a synchronous network where every communications share a common time period T. Each link of E is directed and will have an activation period during T, corresponding to a set of time slots during which the communication can occur without wireless contentions. To ensure transmissions, a collision avoidance method must be chosen as an interference model. Many interference models have been introduced for wireless networks, approximating the physical reality. Some are taking into account MAC layer protocols such as CSMA/CA with RTS-CTS. They are called binary models, meaning that two nodes can either always or never transmit simultaneously unless they communicate together [11]. They can be represented by a conflict graph $G_c = (V_c, E_c)$, where a node in V_c corresponds to a link e of E in G, and there exists a link between two nodes of V_c if and only if the corresponding links in G are interfering. Some other models try to fit a physical layer reality of Signal-and-Interference-to-

Noise-Ratio (SINR) [6]. In this case, the success of receiving a transmission depends on the strength of the signal compared to the level of interference caused by simultaneous transmitting nodes added to the ambient noise.

Given this interference model, a set s is defined as a set of non-interfering links of G. This set is called a *configuration* and corresponds to transmissions that can be active at the same time without interfering with each other. In the conflict graph, a configuration can actually be represented by an independent set of the graph.

The routing and scheduling problem studied here jointly route the packets from routers to gateways and schedule transmissions at rates closed to the maximum throughput capacity. To ensure fairness, the global objective is maximizing the amount of flow that can be guaranteed to each router. Or, given MRs demands, another objective seeks to minimize time period length, leading to a better use of the transmission capacity.

3 Joint Routing and Scheduling Problem Formulations

The goal of the joint routing and scheduling problem is double : it has to find a set of paths on which it can route datas, and ensure that transmissions are free of interferences on each hop of the paths.

In next section, we present a node/arc formulation of the problem, before describing the path/configuration formulation used for column generation.

3.1 Node/Arc Formulation

This initial formulation does not include configurations. Only links and nodes are considered, with constraints and variables related to a *multi-commodity flow formulation*. To prevent interferences, one defines a set $\mathcal{I}(e)$ for each link $e \in E$ containing all links e' interfering with e.

Let f_e^r be the amount of flow sent by router r of V_r on link e, and a_e^t the binary variable saying if link e is active during time slot $t \in [1, T]$, the objective is to guaranty λ units of data per nodes.

$$\max \lambda \tag{1}$$

$$\begin{aligned}
\sum_{e \in \Gamma^+(v)} f_e^r - \sum_{e \in \Gamma^-(v)} f_e^r &= 0, \forall\, r \in V_r,\ v \in V_r \setminus \{r\} && (2)\\
\sum_{e \in \Gamma^-(r)} f_e^r &= 0, \forall\, r \in V_r && (3)\\
\sum_{e \in \Gamma^+(r)} f_e^r &\geqslant \lambda, \forall\, r \in V_r && (4)\\
\sum_{e \in \Gamma^+(g)} \sum_{r \in V_r} f_e^r &= 0, \forall\, g \in V_g && (5)\\
a_e^t + a_{e'}^t &\leqslant 1,\ \forall e \in E, e' \in \mathcal{I}(e), t \leqslant T && (6)\\
\sum_{r \in V_r} f_e^r &\leqslant \sum_{t \leqslant T} a_e^t,\ \forall e \in E && (7)
\end{aligned}$$

Constraints (2)-(5) describe the flow conservation constraints. A MR sends λ units of flow, forwarded by other MRs, until they are received by MGs. (6) claims that a link e cannot be active at a time t in conjunction with a link $e' \in \mathcal{I}(e)$, following the definition of the interference set $\mathcal{I}(e)$ presented above. (7) are the link capacity constraints. The total flow on a link cannot exceed its global capacity on the time period, which is the number of slots the link is activated.

This problem is actually a mixed-integer linear program with binary variables a_e^t. This formulation generates thousands of constraints and variables which make the problem intractable for topologies with more than 20 nodes. Some optimal results have been found and are presented in [4].

In order to tackle larger topologies, a new formulation is derived from this one which allow us to use the column generation method to compute solutions.

3.2 Path/Configuration Formulation

This problem formulation can actually have two different objectives. The first one aims to minimize the time perid length, whereas the second objective wants to maximize node throughput with fairness, saying maximize the minimum outgoing flow at each source node. These two functions are related since having a long time period ensure more link activation and therefore better throughput. But increasing time period length will reduce transmission rate in a permanent network. Thus the goal is quite similar and can be characterized as a network throughput maximization.

Given a set of paths $\mathcal{P}$ and a set of configurations $\mathcal{S}$, f_p represents the amount of flow going through path p of $\mathcal{P}$ and a_s is the activation time of configuration s of $\mathcal{S}$, saying the number of time slots assigned to s. Recall that $\sum_{s\in\mathcal{S}} a_s$ equals the total time period since only one configuration is active during a slot.

The *path-configuration linear formulation*, with two different objective functions described before, is the following :

Master program (MP):

$$\max \lambda \qquad (8) \qquad\qquad \min T \qquad (9)$$

$$\begin{aligned}
\sum_{p\ni e} f_p &\leqslant \sum_{s\ni e} a_s,\ \forall e\in E & \qquad \sum_{p\ni e} f_p &\leqslant \sum_{s\ni e} a_s,\ \forall e\in E & (10)\\
\lambda &\leqslant \sum_{p\in\mathcal{P}_u} f_p,\ \forall u\in V_r & \lambda &\leqslant \sum_{p\in\mathcal{P}_u} f_p,\ \forall u\in V_r & (11)\\
\sum_{s\in\mathcal{S}} a_s &\leqslant T, & \sum_{s\in\mathcal{S}} a_s &\leqslant T, & (12)\\
f_p,\ a_s,\ \lambda &\in \mathbb{N},\ \forall p\in\mathcal{P}, s\in\mathcal{S} & f_p,\ a_s,\ T &\in \mathbb{N},\ \forall p\in\mathcal{P}, s\in\mathcal{S} & (13)
\end{aligned}$$

Constraint (10) verify that flow can be sent on a path only if links composing it are enough active during time period $[1, T]$. Constraints (11) and (12) respectively set the bounds for minimum outgoing flow at each source, and time period length. Therefore objective is either to maximize the minimum outgoing flow (8), or to minimize time period length (9). Depending on the objective chosen, then either T or λ becomes a fixed parameter of the problem.

Problem with objective (9) is not applicable to the node/arc formulation since the configuration characteristics are included in the problem. Constraint (6) actually computes a configuration at each time slot t. It is therefore impossible to deal with a node/arc formulation minimizing time period length T.

3.3 Column Generation

The idea of applying column generation method comes from the exponential size of variables of the problem. It is known that the number of possible paths between mesh routers and gateways is exponential and so is the set of configurations.

Therefore we first solve the continuous relaxation of the master problem with restricted sets $\mathcal{P}_0$ and $\mathcal{S}_0$. If there exists a feasible solution of this restricted master problem, current optimal primal and dual solutions are obtained. Then, using the current optimal dual variables, subproblems seek to generate new paths and configurations which violate the dual constraints and lead to a better solution of the intial problem. These columns found are added to the master problem that is solved again until no such column exists.

In our case, the dual version of the master problem involves three constraints corresponding to the three variables defined in the master program. Introducing the dual variables y_e for links associated with constraint (10) of (MP), x_u for nodes of V_r associated with constraint (11) of (MP), and ω associated with constraint (12), then the dual constraints can be described as follows :

$$\sum_{e \in p} y_e \geqslant x_{\mathcal{O}(p)}, \ \forall p \in \mathcal{P} \tag{14}$$

$$\omega \geqslant \sum_{e \in s} y_e, \ \forall s \in \mathcal{S} \tag{15}$$

$$1 \geqslant \omega \tag{16}$$

where $\mathcal{O}(p)$ denotes source node of path p.

The problem is now to determine if there exists new paths and configurations that make the objective function increase.

4 Auxiliary Programs

As the dual program constraints are defined for paths and configurations, we then have two different auxiliary problems, one for new paths generation and another one for new configurations generation. Auxiliary problems determine if there exists a new column improving the objective value of the master program. If solutions of the sub-problems don't violate the dual constraints, then the corresponding primal solution is optimal.

4.1 The Shortest Path Problem

This auxiliary problem is actually a shortest path problem where edge costs are functions of dual variables of the master problem. Given the optimal dual variables found by solving the contiuous relaxation of the master program with the current basis, the auxiliary path problem is the following :

Definition 1 *Given a weight function $y : E \rightarrow \mathbb{R}^+$, the* Minimum Weighted Path Problem *consists in finding a path $p \in \mathcal{P}$ for wich*

$$y(p) = \sum_{e \in p} y_e$$

is minimum.

We seek to minimize the total path cost $y^*(p)$ and verify if this new path violates constraint (14), saying if $y^*(p) < x_{\mathcal{O}(p)}$. If it does, then an improving path has been found and is added to the current basis of the master problem. As this auxiliary problem is based on shortest path problem, it can therefore be computed efficiently.

Different algorithms have been investigated and compared in terms of time consumption.

4.1.1 Multiflow Formulation

The first approach uses a multiflow formulation, each source sending one unit of flow in the graph. The goal is to find the shortest path between each router source and a gateway. As we don't know exactly the destination node, but only a subset of possible destinations, we introduce a binary variable k_v^r for each $r \in V_r$ and $v \in V_g$, saying if gateway v receives the unit of flow sent by r or not. Let f_e^r be the flow variables on link e from router r, the formulation is the following :

$$\min \sum_{r \in V_r} \sum_{e \in E} y_e f_e^r \tag{17}$$

$$\sum_{e \in \Gamma^+(v)} f_e^r - \sum_{e \in \Gamma^-(v)} f_e^r = \begin{cases} 1 & \text{if } v = r \\ -k_v^r & \text{if } v \in V_g \\ 0 & \text{if } v \in V_r \setminus \{r\} \end{cases}, \ \forall r \in V_r \tag{18}$$

$$\sum_{g \in V_g} k_g^r = 1, \ \forall r \in V_r \tag{19}$$

$$k_i^r, \ f_e^r \in [0,1], \ \forall i \in V_g, r \in V_r, e \in E \tag{20}$$

Since in some cases only few dual variables y_e are non-zero, we have to prevent cycles in solutions computed, otherwise the program can add an important number of *free* arcs with associated weight of 0 in the path solution. We therefore add a small ϵ value at every link weight. But we have to be carefull with this ϵ since we don't want to create better paths with weight $\sum_{e \in p} (y_e + \epsilon)$ than *free* paths with weight $\sum_{e \in p} \epsilon$. Thus we choose $\epsilon = \frac{\min_{e \in E} y_e}{2n}$, with $\min_{e \in E} y_e > 0$ and $n = |V|$, which ensures that $\sum_{e \in p} \epsilon < y_e, \ \forall e \in p$.

Actually, this problem is solved with continuous variables f_e^r and k_v^r. If solution found is a unique path for a given source node, then flow on it equals 1 and we get the same result as in the integer case. Otherwise, this means that all paths have the same continuous weight. This can be derived from the following lemma :

Lemma 1 *Given a set of paths $\mathcal{P}_{s,t}$ between nodes s and t, and a function $f : \mathcal{P}_{s,t} \to \mathbb{R}_+$ with $f(p) = \sum_{e \in p} f_e^s$ and $f(e) = \sum_{p \ni e} f(p)$. If*

$$f^* = \mathit{argmin}_f \sum_{e \in E} (y_e + \epsilon) f(e),$$

then $\forall p \in \mathcal{P}_{s,t}$ with $f^*(p) > 0$,

$$p = \mathit{argmin}_{\overline{p} \in \mathcal{P}_{s,t}} \sum_{e \in \overline{p}} y_e.$$

Proof. Suppose $\exists p_1, p_2$ such that $f^*(p_1) > 0$, $f^*(p_2) > 0$ and $\sum_{e\in p_1} y_e > \sum_{e\in p_2} y_e$. Then $\sum_{e\in p_1}(y_e+\epsilon) > \sum_{e\in p_2}(y_e+\epsilon)$.

Let $f' = \begin{cases} f^* & \forall p \neq p_1, p_2 \\ 0 & \text{on } p_1 \\ f^*(p_1)+f^*(p_2) & \text{on } p_2 \end{cases}$. Thus :

$$\begin{aligned}
\sum_{e\in E}(y_e+\epsilon)f'(e) &= \sum_{e\notin p_1,p_2}(y_e+\epsilon)f^*(e) + (f^*(p_1)+f^*(p_2))\sum_{e\in p_2}(y_e+\epsilon) \\
&< \sum_{e\notin p_1,p_2}(y_e+\epsilon)f^*(e) + f^*(p_1)\sum_{e\in p_1}(y_e+\epsilon) + f^*(p_2)\sum_{e\in p_2}(y_e+\epsilon) \\
&< \sum_{e\in E}(y_e+\epsilon)f^*(e)
\end{aligned}$$

This leads to a contradiction since $f^* = \text{argmin}_f \sum_{e\in E}(y_e+\epsilon)f(e)$. □

4.1.2 Dijkstra Algorithm

The other approach uses Dijkstra algorithm. Our goal is, for each router source, to get the shortest path to one gateway node from V_g. As Dijkstra algorithm computes the shortest path from a single source to all other nodes, if we run it for each node of V_r, it would be time consuming, and it will compute paths between two routers that we don't care about. To optimize the computation, we change slightly the parameters. Given the symmetric property of the connectivity graph G, we transform it into a new graph $G' = (V' = V \cup \{g\}, E' = E \cup E_g)$ as depicted in Figure 2. Let g be a virtual node linked to gateway nodes by arcs $E_g = \{(g,u),\ \forall u \in V_g\}$ with respective weight of 0. Weights for links of E are assigned to the weight of their opposite arc in G, e.g. the dual variable y. Thus if we apply Dijkstra algorithm to compute shortest paths from node g to every other nodes in G', we will obtain every path from nodes of V_r to nodes of V_g in one iteration, without computing useless paths. More precisely, we will get (g,v)-paths for every v of V_r that minimize $\sum_{e'\in E'} y_{e'}$ which corresponds to (v,i)-paths with $i \in V_g$ minimizing $\sum_{e\in E} y_e$.

4.2 The Maximum Independent Set Problem

This problem consists in finding a configuration violating constraint (15). It can be defined into the following problem :

Definition 2 *Given a weight function $y : E \to \mathbb{R}^+$, the* Maximum Weighted Configuration Problem *consists in finding a configuration set $s \in \mathcal{S}$ for wich*

$$y(s) = \sum_{e\in s} y_e$$

is maximum.

As we said before, a set of non interfering links corresponds to a stable set of the conflict graph associated with the network. Thus generating a maximum weighted configuration is actually finding a maximum weighted independent set. This problem is know to be NP-hard, therefore its computation is not efficient and can be exponentional in time. We therefore investigated different approaches for this sub-problem, analysing and comparing an optimal program resolution to the use of a heuristic technique.

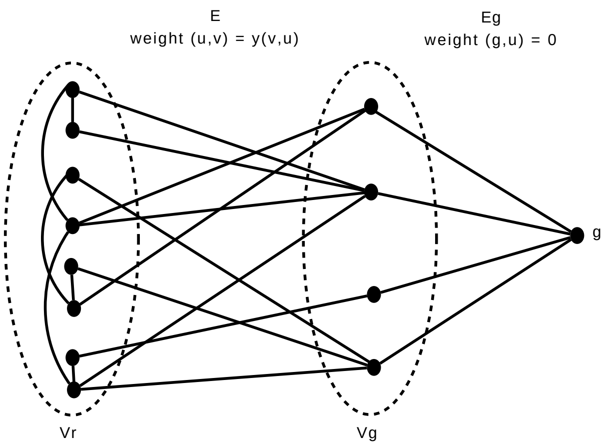

Figure 2: Modified Graph G'

In terms of complexity, we have the following interesting result derived from the theorem of separation and optimization [5, 9]:

Proposition 1 *If there exists a (polynomial-time) ρ-approximation for the* Maximum Weighted Configuration Problem*, then there exists a ρ-approximation for the scheduling problem.*

Proof. In order to solve the dual problem, we only need to separate it. So, given metric y of the dual, we need to decide if it is feasible to output a violated constraint or not. Since to check feasibility means to verify that y is positive, the problem reduces to check the constraints

$$\forall s \in \mathcal{S},\ y(s) \leqslant \omega.$$

For this purpose, one only needs to find a maximum weighted configuration s_0. If its weight is strictly larger than ω then we output $y(s_0) \leqslant \omega$ as violated constraint, otherwise y is feasible. Therefore, if such s_0 can be found efficiently, the result follows (for the exact case). Let us assume now that we have a ρ-approximation. Then, it provides us with a configuration s_1 such that $y(s_1) > \omega$ in polynomial time, or otherwise we know that the metric y/ρ is feasible, and the approximation case follows. □

When a new configuration s has been found by the sub-problem, we then look if its corresponding weight $y^*(s)$ is greater than ω. If it does, this means that constraint (15) is violated and that s improves the current solution.

4.2.1 Optimal Formulations

To optimaly generate new configurations sets, we use a binary program already presented in [17], [2] and [3]. It also depends on dual variables associated with links, and as for the previous auxiliary problem for paths, a new variable z_e has been introduced for links to determine if link e will be added to the new configuration or not.

$$\max \sum_{e \in E} y_e z_e \tag{21}$$

$$z_e + z_{e'} \leqslant 1,\ \forall e \in E,\ e' \in \mathcal{I}(e) \quad (22)$$
$$z_e \in \{0,1\},\ \forall e \in E \quad (23)$$

Recall that $\mathcal{I}(e)$ corresponds to the set of links interfering with e. This formulation uses the binary interference model presented in section 2 of this paper. But other interference models can be applied like the *SINR model* where a communication between two nodes i and j can occur if the *Signal-to-Noise-and-Interference-Ratio* at node j ($SINR_j$) is greater then a fixed threshold γ :

$$SINR_j = \frac{p(i,j)G(i,j)}{\eta + \sum_{(k,l)\neq(i,j)} p(k,l)G(k,j)}.$$

To consider this model, one has to replace constraint (22) by the following ones :

$$p(i,j)G(i,j)z_{(i,j)} \geqslant \gamma \cdot (\eta + \sum_{(k,l)\neq(i,j)} p(k,l)G(k,j)z_{(k,l)}), \forall (i,j) \in E \quad (24)$$
$$\sum_{e \in E_i} z_e \leqslant 1, \forall i \in V \quad (25)$$

Unfortunately these are integer programs that cannot be relaxed since interfering links could be chosen in the same configuration. Thus optimal resolution is time consuming in the column generation process and must be replaced by approximation algorithms to be more efficient.

4.2.2 Approximation Algorithm

Here we decide to compute a greedy algorithm to find various maximal for inclusion weighted configurations. As several configurations may violate constraint (15), adding only one configuration by iteration may increase the total number of iterations and therefore the resolution time. Thus we decide to develop an algorithm which computes quickly a fixed number of maximal weighted configurations (see Algorithm 1). This al-

Algorithm 1 : Maximal Weighted Configurations Generation

Sort arcs e of E such that e_1 is before e_2 if $y_{e_1} \geqslant y_{e_2}$
$Result \leftarrow \emptyset$
$nbIt \leftarrow 1$
while $nbIt \leqslant 10$ **do**
  $s_{nbIt} \leftarrow \{e_{nbIt}\}$
  for $i = nbIt$ to $|E|$ **do**
    if e_i does not interfere with elements of s_{nbIt} **then**
      $s_{nbIt} \leftarrow s_{nbIt} \cup \{e_i\}$
    end if
  end for
  $Result \leftarrow Result \cup s_{nbIt}$
end while

gorithm permits to compute rapidly approximated maximum weighted configurations, and to use optimal programs only when it does not find solutions that violate the dual constraint anymore. This significantly reduces resolution time.

5 Tests

Tests have been realized using the MASCOPT[1] library developed by team members, and ILOG CPLEX.

5.1 Validation

First, we validate our path/configuration formulation with small topologies that can be solved optimaly. We compare the continuous results from the column generation applied on the relaxation of the path/configuration formulation with the integer optimal results of the node/arc initial problem. Results are depicted on figure... and validate the formulation. Overall, if we solve the continuous relaxation of the node/arc formulation (except for binary variables a_e^t), then results are the same than those from column generation where we apply branch and bound on configuration variables a_s.

5.2 Simulations

As bigger topologies can be solved optimaly because the thousand constraints and variables generated by the problem, we only solved them with our column generation method. Tests have been realized on a set of random instances. n points are deployed on a plane of length 1 and height $1/4$, following a Poisson process (law?). A transmission radius is fixed in order to get a connected graph with mean degree $\overline{d} = \max(5, \frac{n}{10})$. Gateways are uniformly and randomly chosen among the nodes. Other nodes becomes routers with a unitary demand to send.

We generate topologies of size between 10 and 100. For each topology, we have tested with different number of gateways from 1 to $n/\overline{d}$. In the last case, gateways only communicate with their neighbors.

By normalizing distances between the nodes in order to obtain a transmission radius of 1, we obtain poissonian graphs of density $\frac{n}{10\pi}$. Graphs are then locally dense, but, due to the rectangular property of the plane considered, they stay spread.

In the following we present results obtained for the theoretical capacity evaluation of the network and analyse them. We notably discuss on complexity results.

5.2.1 Impact of parameter T

The time period length T is given as an input of our model. It has an impact on the network throughput because every active path where λ units of flow are sent during $[1, T]$ will have a flow rate of λ/T in permanent regime.

Another impact concerns problems complexity. In the node/arc formulation, there is a binary variables per arc and per slot, leading to an ordered link activation over time period. On the contrary, the path/configuration formulation does not specify during which slot a link is activated.

A consequence is when parameter T is increased, then the node/arc formulation gets more variables whereas the path/configuration formulation as always the same number of variables. The column generation resolution time will not grow as fast as the node/arc problem.

[1]http://www-sop.inria.fr/mascotte/mascopt/

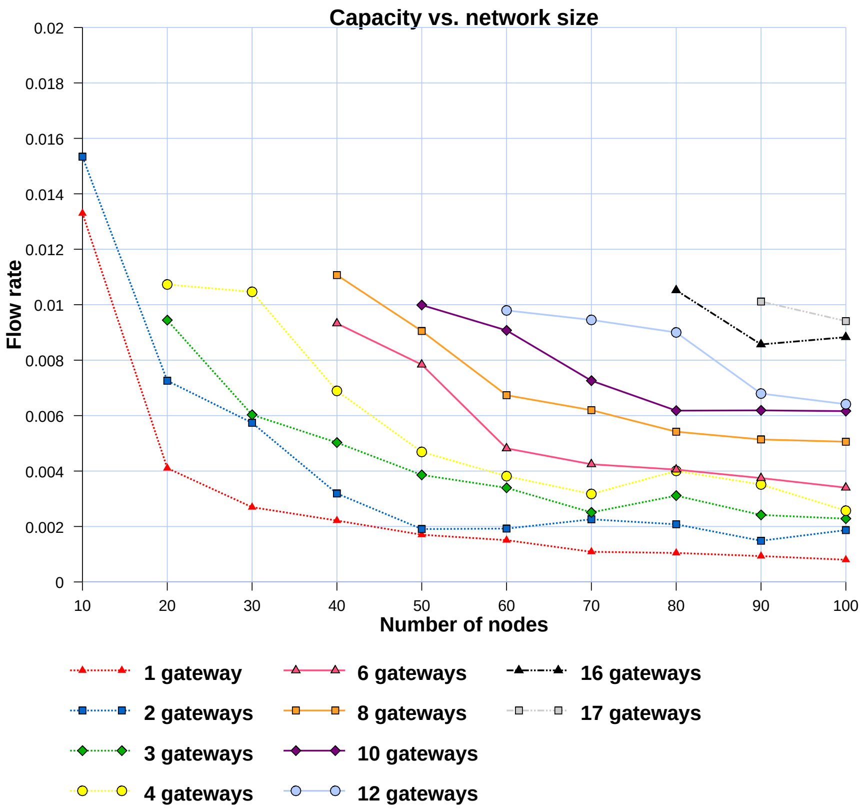

Figure 3: Capacity decreases proportionnaly as the network size increases.

5.2.2 Capacity and network size

In a data gathering environment with one gateway, works have shown that the per-node capacity is decreasing proportionnaly as the network size increases [9]. We suppose that the same phenomenon occurs with various gateways.

Figure 3 confirms our guess. Deeper investigations would show that, outside of a contention area located around each gateway, the linear program is not really constrained.

Then, it would be interesting to develop efficient algorithms that only optimize around the gateways. Farther areas would thus be treated by fast and unprecise processes (for instance greedy algotrithms). This assumption is supported by the complexity of the independent set computation that is sub-linear with the network size, but depends on the gateways density (i.e. to the maximal size of the contention areas).

5.2.3 Capacity and gateways density

The link between capacity and gateways density is not clear so far. Linear dependency corresponds to an obvious upper bound, but interferences between gateways could degrade performances. To study this phenomenon, we have generate different random

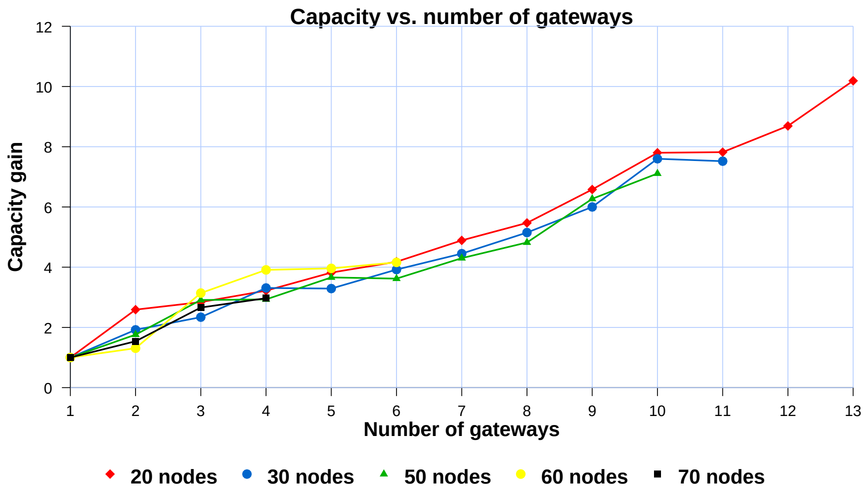

Figure 4: Capacity increases sub-linearly as the number of gateways grows. (Capacity is normalized by its value with 1 gateway)

gateways placements for each topology. Figure 4 presents the mean gain obtained by adding new gateways. Values are normalized by the rate obtained with one gateway.

Results show a linear gain in the number of gateways with a gradient slightly lower than 1. With a correct placement, even non optimized, wireless mesh networks act efficiently. A more detailed analysis would have shown that the placement has a strong impact on the network performances. When two gateways are too closed from each other, they actually act as only one. Indeed, interferences produces by both gateways are too strong and prevent a simultaneous traffic collect.

We can see an exemple on grids where we place two gateways oppositely **??**.

We conjecture that a minimum distance is not even necessary but also sufficient to a correct traffic flow. This distance could be explicitely computed on regular topologies.

6 Conclusion

In this work, we have presented a cross-layer optimization-based method for the joint routing and scheduling problem for Internet-providing wireless mesh networks. We developped exact formulations using linear programming and column generation. This approach allows to compute solutions for large scale networks and study the capacity offered to clients upon different parameters.

Simulations highlight the fact that the gateways placement has to be carefully done because it has a serious impact on the network performances. However, tightly optimizing the placement seems to be irrelevant. A combinatorial study should show the existence of a critical area centered at the gateways beyond which the problem can be roughly solved.

These ideas can have an impact on the capacity of wireless mesh networks as much as on the complexity of their design.

7 Conclusion

In this work we have presented the joint routing and scheduling problem for wireless mesh networks under a path/configuration formulation. It permits the use of column generation to iteratively find new improving paths and configurations, leading to an optimal solution of the continuous relaxation of the problem.

Simulations have been realized on random and grid topologies with various number of gateways. Resolution time is promising but still depends on the auxiliary configuration problem known to be NP-hard. Use of approximation schemes to generate maximal independent sets are hardly recommended.

References

[1] I.F. Akyildiz, X. Wang, and W. Wang. Wireless mesh networks: a survey. *Computer Networks*, 47(4):445–487, 2005.

[2] G. Carello, I. Filippini, S. Gualandi, and F. Malucelli. Scheduling and routing in wireless multi-hop networks by column generation. In *INOC 2007*, April 2007.

[3] C. Gomes and G. Huiban. Multiobjective analysis in wireless mesh networks. In *IEEE MASCOTS*, October 2007.

[4] C. Gomes and H. Rivano. Fair joint routing and scheduling problem in wireless mesh networks. Research Report 6198, INRIA, 05 2007.

[5] Martin Grötschel, László Lovász, and Alexander Schrijver. The ellipsoid method and its consequences in combinatorial optimization. *Combinatorica*, 1(2):169–197, 1981.

[6] P. Gupta and P.R. Kumar. The capacity of wireless networks. *IEEE Transactions on Information Theory*, 46(2):388–404, March 2000.

[7] K. Jain, J. Padhye, V. Padhamanabhan, and L. Qiu. Impact of interference on multi-hop wireless network performance. In *ACM MobiCom*, pages 66–80, September 2003.

[8] J. Jun and M.L. Sichitiu. The nominal capacity of wireless mesh networks. *IEEE Wireless Communications*, 2003.

[9] R. Klasing, N. Morales, and S. Pérennes. On the complexity of bandwidth allocation in radio networks with steady traffic demands. Research report, INRIA/RR-5432 and I3S/RR-2004-40-FR, 2004.

[10] M. Kodialam and T. Nandagopal. On the capacity region of multi-radio multi-channel wireless mesh networks. In *First IEEE WiMesh*, September 2005.

[11] Anil Kumar, Madhav Marathe, Srinivasan Parthasarathy, and Aravind Srinivasan. Algorithmic aspects of capacity in wireless networks. In *ACM SIGMETRICS*, volume 33, pages 133–144, 2005.

[12] Sriram Lakshmanan, Karthikeyan Sundaresan, and Raghupathy Sivakumar. On multi-gateway association in wireless mesh networks. In *IEEE WiMesh*, pages 135–137, September 2006.

[13] Hengchang Liu and Baohua Zhao. Optimal scheduling for link assignment in traffic-sensitive stdma wireless ad-hoc networks. In *Proceedings of the 3rd International Conference on Networking and Mobile Computing (ICCNMC 2005)*, volume 3619 of *Lecture Notes in Computer Science*, pages 218–228, Aug 2005.

[14] Marco E. Lübbecke and Jacques Desrosiers. Selected topics in column generation. *Operations Research*, 53(6):1007–1023, 2005.

[15] W.-H. Tam and Y.-C. Tseng. Joint multi-channel link layer and multi-path routing design for wireless mesh networks. In *IEEE INFOCOM*, May 2007.

[16] P. Wang and S. Bohacek. Toward tractable computation of the capacity of multi-hop wireless networks. In *IEEE INFOCOM*, May 2007.

[17] J. Zhang, H. Wu, Q. Zhang, and B. Li. Joint routing and scheduling in multi-radio multi-channel multi-hop wireless networks. In *IEEE BROADNETS*, pages 678–687, October 2005.

Centre de recherche INRIA Sophia Antipolis – Méditerranée
2004, route des Lucioles - BP 93 - 06902 Sophia Antipolis Cedex (France)

Centre de recherche INRIA Futurs : Parc Orsay Université - ZAC des Vignes
4, rue Jacques Monod - 91893 ORSAY Cedex
Centre de recherche INRIA Nancy – Grand Est : LORIA, Technopôle de Nancy-Brabois - Campus scientifique
615, rue du Jardin Botanique - BP 101 - 54602 Villers-lès-Nancy Cedex
Centre de recherche INRIA Rennes – Bretagne Atlantique : IRISA, Campus universitaire de Beaulieu - 35042 Rennes Cedex
Centre de recherche INRIA Grenoble – Rhône-Alpes : 655, avenue de l'Europe - 38334 Montbonnot Saint-Ismier
Centre de recherche INRIA Paris – Rocquencourt : Domaine de Voluceau - Rocquencourt - BP 105 - 78153 Le Chesnay Cedex

Éditeur
INRIA - Domaine de Voluceau - Rocquencourt, BP 105 - 78153 Le Chesnay Cedex (France)
http://www.inria.fr
ISSN 0249-6399