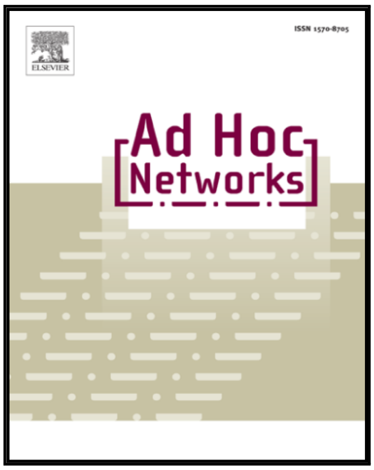
Accepted Manuscript



Improving Load Balancing, Path Length, and Stability in Low-Cost Wireless Backhauls

Micael O.M.C. de Mello, Vinicius C.M. Borges, Leizer L. Pinto, Kleber V. Cardoso

PII: S1570-8705(16)30124-X
DOI: 10.1016/j.adhoc.2016.05.002
Reference: ADHOC 1388

To appear in: *Ad Hoc Networks*

Received date: 1 July 2015
Revised date: 6 May 2016
Accepted date: 10 May 2016

Improving Load Balancing, Path Length, and Stability in Low-Cost Wireless Backhauls

Micael O. M. C. de Mello, Vinicius C. M. Borges*, Leizer L. Pinto, Kleber V. Cardoso

Instituto de Informática
Universidade Federal de Goiás (UFG)
Goiânia – GO – Brasil

Abstract

Futuristic wireless networks are being proposed to promote a significant improvement in performance, mainly in highly dense scenarios. However, cost constraints are also a key concern for the next generation of wireless networks. In this context, Low-cost Wireless Backhauls (LWBs) can provide relevant contributions. LWBs are based on WLAN technologies, such as Wireless Mesh Networks, in which gateways to the wired networks are potential bottlenecks, and load balancing is critical for performance. In spite of several proposals to deal with load balancing, they fail to combine three key aspects: (1) stability, (2) reduction of the average path length, and (3) throughput fairness. Our proposal addresses these aspects by adopting a joint approach for routing and channel assignment. The routing part is composed of a heuristic that employs an on-demand local solution in which load balancing is combined with the three aforementioned key aspects. We have carried out an in-depth simulation study in ns-3 to assess the impact of our proposal on traffic performance when compared with the state of the art. Our proposal is superior in most of the scenarios, in particular in terms of stability and average path length. Our proposal also increases the aggregate throughput and minimum throughput per flow in the network, especially when the number of flows is large.

*Corresponding author

Email addresses: micaelmello@inf.ufg.br (Micael O. M. C. de Mello), vinicius@inf.ufg.br (Vinicius C. M. Borges), leizer@inf.ufg.br (Leizer L. Pinto), kleber@inf.ufg.br (Kleber V. Cardoso)

Preprint submitted to Ad Hoc Networks *May 11, 2016*

Keywords: Low-cost wireless backhauls, load-balancing routing, joint routing and channel assignment, path length, routing stability

1. Introduction

After 2020, a new generation of wireless communications, a.k.a. 5G Networks, is expected to support a 1000-fold increase in the present traffic demand [1] and to provide unprecedented ubiquity. Several new technologies [2] are under investigation or are being designed to address these expectations, for example, interference cancellation, millimeter wave, massive MIMO, and visible light communication. On the other hand, 5G networks will also be under considerable pressure to introduce budget constraints [1, 3, 4], since companies cannot count on increases in the users' bills to cover the large expenditure required for the development and deployment of new technologies. This situation encourages the adoption of approaches such as infrastructure and spectrum sharing [5] and heterogeneous networks [3, 6] for a more efficient use of resources. WLANs are one of the most important technologies in heterogeneous networks, particularly for cellular offloading and D2D communications [7].

Another promising approach for 5G networks is *network densification* [8], which has the potential to combine new technologies with cost-effectiveness. Basically, network densification consists of combining spatial densification and spectral aggregation. Spatial densification means increasing the number of wireless infrastructure nodes that, for example, provide connectivity for mobile users and IoT devices. Both spatial densification and spectral aggregation entail increasing the amount of traffic between the Internet and the nodes of the wireless infrastructure. Thus, the impact of network densification depends on the densification of the backhaul. While fiber optics and some recent wireless technologies (e.g., millimeter wave) may offer excellent results in terms of performance, their costs are, in some key scenarios, too high. For example, IoT sensors and actuators are inexpensive devices that are being deployed in large numbers in many public places. Thus, there is a huge demand for coverage involving low-power devices, and the aggregate throughput demand can be supported by low-cost wireless backhauls based on WLAN technologies. In developing countries, such as Brazil, India, and South Africa, there are a lot of poor dense communities, which are getting access to cheap wireless mobile devices (e.g., smartphones and tablets) but are unable to pay for conventional wireless data plans. Low-cost wireless backhauls (LWBs), which are also based on WLAN technologies, can be a method that enables companies to extend their infrastructures to these communities and offer affordable wireless data plans.

Low-cost wireless backhaul can be seen as a Wireless Mesh Network (WMN) [9] with some specific features. Like a WMN, an LWB comprises gateways, routers, and clients. A set of routers offers a multi-hop backbone to reach the gateways, which usually have a wired connection to an external network or the Internet. On the other hand, WMNs are often deployed by communities or small companies [10, 11], where cost is a critical issue. Thus, WMNs are generally built with low-cost off-the-shelf hardware [12], use a very low number of interfaces per device (e.g., there are hardly ever more than three air interfaces), and employ ad hoc topologies where it is common to find bridging nodes [13]. LWBs are networks that are designed to be deployed by mobile network operators (MNOs) to increase their coverage at a low cost. However, the alternative solutions available for the MNOs are commonly expensive (e.g., fiber optics), thus high-quality customized hardware and a larger number of radios (e.g., five air interfaces) per device are acceptable. In addition, the deployment of LWBs tends to be planned, and uses dense topologies that increase the potential network capacity and avoid the need for bridging nodes. This context requires solutions that employ the resources efficiently, and exploit the potential of multiple paths for improving network capacity.

In the context of WMNs, it is well known [14] that networks comprising routers with multiple radios and multiple channels can significantly increase the aggregate network capacity. In this regard, several papers have sought solutions for the MAC layer [15, 16, 17] and for the network layer [18, 19, 20], i.e., those that involve routing protocols. However, the most promising suggestions [21, 22, 23] combine information from both layers and thus can significantly benefit from the multiple available paths. The main problem in the MAC layer is the channel assignment, while load balancing routing is the most important problem in the network layer. Additionally, the routing decision should take into account the need to reduce the number of hops between the source and destination, so as to shorten the end-to-end delay and improve the contention-based media access [24, 25]. Recent publications [24, 25] have evaluated their proposals under different conditions and made use of more realistic traffic profiles. This is the starting point of our paper.

We noticed that one of the most promising approaches [24], based on joint routing and channel assignment, still has some serious drawbacks when employed in LWBs: it does not achieve a satisfactory trade-off between load balancing and path length, which mainly affects flows with long paths. In addition, it requires a large number of channel re-assignments, which significantly increases computational costs, due to the complexity of the channel assignment algorithm. This problem led to the development of our joint solution, called the Joint approach for Improving Load balancing and Path length (JILP) in LWBs. The main contributions of our solution are in the area of routing. JILP employs a set of simple and efficient load balancing algorithms, called Bottleneck, Path Length and Routing overhead (BPR) [26], the purpose of which is to add and remove routes for the flows. If a bottleneck increases, the algorithms only operate on the flows going through it, either by seeking to reduce the bottleneck or finding new routes with shorter paths.

In the present paper, we provide a more formal and detailed description of the BPR algorithms, and also carry out a more in-depth evaluation of performance that involves both parts of our joint solution: routing and channel assignment. In [26] we addressed the topological and overhead issues of the routing part. In other words, we did not investigate performance metrics such as throughput and fairness, and the evaluation was restricted to aspects of the routing algorithms. In the present paper, we have expanded the evaluation to investigate our solution in greater depth and also compare it with another state of the art joint approach [24], conducting a large number of simulations. On the basis of these simulation results, we show that, when compared with [24], JILP keeps the network bottleneck close to the optimum while reducing the average path lengths and the number of channel re-assignments. In addition, in most of the evaluated scenarios, JILP increases the aggregate throughput and the fairness between the flows.

The rest of this paper is structured as follows. Section 2 examines some relevant related research. Section 3 outlines the problem formulation for load balancing and path length. Section 4 describes our proposal and gives a detailed description of the algorithms. Section 5 presents the evaluation and discusses the results with regard to the distinct number of flows, channels, and scenarios. Finally, Section 6 summarizes our main findings and gives some suggestions for directions for further research.

2. Related Work

Since LWB is still a new concept, in this section we will review relevant publications related to WMN, and highlight any differences that might arise. Most of the traffic in WMNs tends to cross the network's boundaries through gateways, mainly in the direction from the Internet to the internal network, i.e., it is download traffic. The number of gateways is usually smaller than the number of mesh routers owing to the increase in financial costs. Thus, the wireless links close to the gateways are potential network bottlenecks and thus reduce the throughput of the flows that pass through the bottlenecks because of the unfair distribution of these flows in the mesh routers that are close to the gateways and the high level of contention or self-interference. Load balancing routing and channel assignment play an important role in the traffic performance of a WMN [27]. This section describes the related research into load balancing routing, channel assignment, and the combination of both.

Load balancing routing is one of the main areas researched in WMN and has received considerable attention from the scientific community. The centralized algorithms employed by [28, 29, 30] handle the load balancing routing in a single gateway environment by distributing the traffic load between the routing subtrees of which the gateway is the root. These algorithms have a drawback caused by the routing subtrees, i.e., the individual flow routes are not independent since the routes are restricted by the tree structure. As a result, the tree structure does not allow the load balancing routing algorithms to take full advantage of the dense features of the LWBs.

Many load-aware routing metrics have been designed for WMNs [31] to take advantage of the dense topologies, such as WCETT-LB [18], ILA [19], CWB [20], NLR [32], and LAM [33]. These metrics act as a part of the routing protocols, which means they are distributed solutions for load balancing. Load-aware routing metrics use at least one load measurement (e.g., queue length, channel busy time, and number of flows) to be aware of the congested links or nodes. The routing protocol disseminates the routing information (i.e., the routing metric). On the basis of this information, each node can compute its own routing table by using a routing algorithm (e.g., the shortest path). However, load-aware routing metrics usually have a high level of routing oscillation due to the frequent changes in the traffic load [34], which degrades the performance of many applications. Statistical functions and updated propagation threshold mechanisms have been suggested to smooth the values of the metrics, but these only serve as a palliative solution. Furthermore, load-aware routing schemes also result in longer paths, since the traffic tends to be routed around the congested links or nodes.

Load balancing routing mechanisms were developed by adopting an approach that selects multiple paths for the same source–destination pair [35, 36]. This multipath approach selects routes to transmit a single flow in a parallel way, i.e., packets of the same flow are sent by different paths. Moreover the approach provides a theoretically better load balancing, even though the large amount of out-of-sequence packets has an adverse effect on most of the applications. Congestion control mechanisms of transport protocols (such as TCP) and playout buffer strategies of multimedia applications are not designed to deal with large and regular packet reordering.

In general, load balancing routing is not enough to realize the potential of multiple paths. Wireless links that are close to each other contend for access to the medium, thus it is important to find a solution for channel assignment to mitigate this interference. The use of multiple radios helps to relieve the contention, but it leads to the problem of making an optimum choice of channel assignment for the radios (air interfaces), which is NP-hard [37]. There have been several attempts to find solutions to this problem, since the cost reduction resulting from the use of IEEE 802.11 devices has encouraged the design of dense networks that use nodes with multiple radios. Naturally, the capability of the multi-radio multichannel requires the use of orthogonal channels [15], which enables a better spatial reuse than a single-channel approach.

There are many centralized channel assignment algorithms, which can be classified into three main categories [15]: 1) network modeling based on a colored graph [16, 17, 22, 38], 2) network modeling based on the flows [21, 23, 39], and 3) partition network modeling or clustering [40, 41]. The graph and partition network modeling algorithms share the same objective, which is to reduce the level of contention. The algorithms for network modeling based on flows tend to allocate more orthogonal channels to wireless links that are frequently used, i.e., they are load-aware channel assignment algorithms. There is other related research that proposes distributed protocols and algorithms to assign channels in multi-radio multi-channel networks [42, 43, 44]. In this area, each router performs an instance of the algorithm and makes local decisions based on its own information and on information from its neighbors. In addition, these algorithms employ strategies to select the channel with the lowest load and the lowest level of interference. The centralized algorithms usually achieve a better performance than the distributed ones because they have a global view of the network, but they rely on the network information topology and require mechanisms to disseminate the channel assignment decisions. Moreover, although distributed algorithms act locally and do not depend on all the network information, they need protocols to disseminate information to the neighboring nodes or the nodes must act with some degree of cooperation between them.

The joint approach combines routing and channel assignment strategies that take into account the interdependence and the high coupling between these strategies [23, 45]. On the one hand, the routing seeks to select the paths, which can take into consideration throughput fairness, load balancing, average path length and stability. On the other hand, channel assignment aims to allocate more bandwidth to the overloaded or lower-capacity channels. Hence, the main objective of the joint approach is to find a combination of path selection and channel assignment for the radios in each router so that the network throughput is maximized. Additionally, route stability and fairness between the flows are also desired properties of a joint approach solution. As would be expected, finding an optimum solution for the joint approach problem is also NP-hard [23, 24] and, depending on the problem formulation, there may exist several non-dominated solutions, i.e., the problem may become multi-objective.

A good deal of work has been carried out on the subject of joint routing and channel assignment in WMNs. Raniwala *et al.* [21] proposed a centralized solution in which the channel assignment algorithm allocates orthogonal channels to overloaded links. However, no load balancing solution was used, since the authors preferred to employ the shortest path routing and randomized multipath routing algorithms. In addition, the channel assignment algorithm depends on *a priori* knowledge of the network traffic. Raniwala *et al.* [42] also developed a distributed solution to the joint approach in which the channel assignment algorithm performs local decisions for each node. The adopted approach seeks to assign the least-used channels around the neighborhood. The routing part is based on a solution that makes use of subtrees to perform the load balancing routing. Again, *a priori* knowledge of the network traffic is necessary.

Alicherry *et al.* [23] designed a mathematical model for the joint approach. They employed an algorithm to obtain an approximate solution for the link scheduling and channel assignment to maximize the network throughput. This approach depends on *a priori* knowledge of the network traffic. Gardellin *et al.* [46] suggested a divide-and-conquer solution for a joint approach. This involves dividing the network into smaller groups or clusters, where the optimal solution for each cluster is calculated and then all the solutions are combined. A knowledge of network traffic, the rate allocation, and the received signal strength are required for each radio in each router. Wu *et al.* [25] gives priority to the multi-radio and multi-channel assignment by using nodal interference information to form cliques for inter-clusters and intra-clusters in the network. **Gammar** ***et al.*** **[47] also employs clustering scheme to smooth out interference through the assignment of distinct channels for each cluster, but their solution does not take into account the throughput fairness and average path length**.

Avallone and Di Stasi [48] developed a centralized algorithm for joint routing and channel assignment, called Resilient Directed Acyclic Subgraph (RDAS), that is focused on improving resiliency through load balancing routing. RDAS is based on an MPLS splitting policy which divides flows in multiple paths and seeks to distribute them so that each set of interfering links does not exceed the channel capacity. The authors adopt the hose traffic model, i.e., they have knowledge of the maximum amount of traffic entering or leaving the network at each gateway node. Thus, *a priori* **knowledge of the network traffic is necessary. Although RDAS uses a fixed threshold for the path length, they do not seek to provide an equilibrium between distribution of flows and path length. Additionally, the algorithm does not take into account stability for both routing and channel assignment. Avallone** ***et al.*** **[49] presents a centralized algorithm, called Minimum Power Channel Assignment and Routing Algorithm (MP-CARA), which maximizes power savings and network performance based on properly routing flows, assigning channels to radios and selecting nodes/radios that can be turned off. MP-CARA does not address the average path length and stability**.

The routing algorithm must be aware of a link scheduling mechanism in order to choose a valid radio and channel along the path that can improve the load balancing and congestion. However, depending on the PHY layer specification adopted, the number of non-overlapping channels is very small, which makes it difficult to reduce the contention. There are other joint approaches based on centralized and offline algorithms, such as Gardellin *et al.* [46], Tang *et al.* [50], and Avallone *et al.* [51]. However, all of these suggestions share the same drawback, which is their dependence on an unrealistic or simplified throughput model that requires *a priori* knowledge of the traffic load.

To overcome this limitation, Galvez and Ruiz [24, 52] introduced the Joint Routing, Channel Assignment and Rate allocation Heuristic (JRCAR), which routes and balances the traffic load at the flow level. JRCAR does not depend on a throughput model and can thus adapt to real-time variations in traffic load. JRCAR uses separate heuristics for routing and channel assignment, both of which take into account the load on the wireless links. Initially, JRCAR is responsible for handling the load balancing routing and after that it performs the channel assignment for the overloaded links. The load balancing routing is handled at the application level of the TCP flows: each flow is identified by the source address, destination address, source port, and destination port. Furthermore, JRCAR offers a simple trade-off between load balancing and path length, which has not been taken into account by the previous research. JRCAR employs a stretch factor parameter defined by Gao *et al.* [53], which limits the length of a path between two nodes s and d according to a factor $\tau\eta^{sd}$, where η^{sd} is the length of the shortest path.

JRCAR prioritizes the routing of the flows with the smallest number of candidate paths, and rejects alternatives that have a length greater than $\tau\eta^{sd}$ or that have loops. As a result, the flows with the greatest number of candidate paths are routed last, since these flows might have a higher number of alternative routes. This has a negative effect on the long-path flows. In addition, the algorithms that compute the candidate paths and subsequently the load balancing flow routing have a high computational complexity, which affects their responsiveness to highly dynamic traffic patterns. In spite of these drawbacks, JRCAR is a state of the art approach for joint routing and channel assignment, since it does not rely on *a priori* knowledge of the network traffic. This is the reason why we selected it as the main alternative for comparison with our proposal (JILP).

Table 1 identifies which properties are explicitly addressed by the related works. This table takes into account some of the most common properties: type of approach (centralized or distributed), throughput fairness, stability, and path length.

3. Problem Formulation

In this section, there is a formal description of an optimization problem that involves the following objectives related to routing: balancing the traffic

Table 1: Summary of the Related Works on Joint Routing and Channel Assignment

Reference	Type of approach	Throughput fairness	Stability	Path length
Raniwala *et al.* [21]	centralized	✓	–	✓
Alicherry *et al.* [23]	centralized	✓	–	✓
Galvez and Ruiz [24]	centralized	✓	✓	✓
Wu *et al.* [25]	centralized	✓	–	–
Raniwala *et al.* [42]	distributed	✓	–	✓
Gardellin *et al.* [46]	centralized	✓	–	✓
Gammar *et al.* [47]	distributed	–	✓	–
Avallone and Di Stasi [48]	centralized	–	–	✓
Avallone *et al.* [49]	centralized	✓	–	–
Tang *et al.* [50]	centralized	✓	–	–
Avallone *et al.* [51]	centralized	–	–	✓

flows and reducing the path length of these flows. We also describe some aspects of LWB networks that have a bearing on the problem formulation and its potential value for finding solutions.

The network topology is represented by a directed graph $G = (V, E)$, in which every router is a node. There exists an edge, i.e., a wireless link, between two nodes $s,d \in V$ if node d can receive a transmission from node s. Let e_{sd} denote a directed edge from s to d. We assume the connectivity is symmetric, i.e., edge $e_{sd} \in E$ if and only if $e_{ds} \in E$. Let F denote the set of Internet flows. The source of a flow $f \in F$ is denoted by s_f, while d_f denotes the destination. A flow either originates or terminates at a gateway node. A flow route specifies the gateway employed by the flow. We assume that all gateways are centrally managed, to avoid any ambiguity when establishing the routes.

The problem of load balancing routing is how to select a path for each flow in such a way that the load is evenly distributed over the network. However, most of the routing algorithms for load balancing result in very long paths, since they tend to go around the overloaded edges when making the routing decision, which can degrade the performance of an application [54]. An efficient solution to the load balancing routing problem should establish an equilibrium between load balancing and path length. Thus, this problem can be split into two conflicting objectives: minimizing the overload of the edges (defined as the number of flows that use the edge), and minimizing the path length for every flow.

Given a set of flows F, the topology graph G and the length of the shortest path for each flow $f \in F$, described by η^f, the multi-criterion routing problem with load balancing and path length objective functions is defined as

$$minimize \left\{ max_{e_{sd} \in E} \left\{ \sum_{f \in F} a^f_{sd} \right\} \right\} \tag{1}$$

$$minimize \left\{ \sum_{e_{sd} \in E} a^f_{sd} \right\}, \forall f \in F \tag{2}$$

subject to

$$\sum_{e_{s_f d} \in E} a^f_{s_f d} = \sum_{e_{sd_f} \in E} a^f_{sd_f} = 1, \forall f \in F \tag{3}$$

$$\sum_{e_{sd} \in E} a^f_{sd} = \sum_{e_{ds} \in E} a^f_{ds}, \forall d \in V - \{d_f, s_f\}, \forall f \in F \tag{4}$$

$$\sum_{e_{sd} \in E} a^f_{sd} \leq \Gamma \eta^f, \forall f \in F \tag{5}$$

$$a^f_{sd} \in \{0, 1\}, \forall e_{sd} \in E \text{ and } \forall f \in F \tag{6}$$

where the binary variable a^f_{sd} is equal to one if $f \in F$ traverses e_{sd}, otherwise it is zero. $\Gamma \geq 1$ is a given stretch factor parameter, which makes it possible to control the maximum path length. The two constraints in (3) ensure that the flow f initiates from its source, s_f, and reaches its destination, d_f. Constraints (3), (4), and (6) ensure that the traffic of a flow only follows one path from its source to its destination. As described earlier, this is important for many applications because it avoids the problem of severe packet reordering and fluctuations in the round-trip time of the flows. Constraint (5) ensures that the path length will not be greater than the limit imposed by the parameter Γ, which is important for real-world use even though one of the objectives is minimize the path length. According to this kind of problem formulation, the load balancing routing problem can be classified as a multi-criterion problem, since its objective function (1) seeks to find a solution (routing F flows in G) to minimize the most congested edge, whereas $|F|$ objective functions in (2) seek to reduce the path length for the flow routing.

Since the problem has more than one objective, there is a set of non-dominated solutions, or Pareto optimal solutions [55]. Moreover, this routing problem requires a solution in a short period which means that the time needed to calculate the Pareto optimal solutions is, in general, unacceptably high. Non-exact methods, e.g., heuristics, are a more suitable approach, since they can obtain satisfactory solutions very quickly. In the next section we will describe a heuristic for solving the problem outlined here.

4. Bottleneck, Path length and Routing overhead heuristic (BPR)

BPR is a heuristic to optimize the multi-objective problem described in the previous section. BPR ensures that model constraints 3–6 will be obeyed while it operates dynamically for new and recently finished flows. For each flow that starts, BPR seeks the best route in terms of the network bottleneck (objective (1)) and the path length (objective (2)). For each flow that finishes, BPR seeks to minimize the bottleneck of the network and the path length used by the other flows.

We include some additional notation and definitions for a precise description of the algorithms outlined in this section:

- n: new flow.
- t: finished (or terminated) flow.
- P_f: path used by the flow f.
- F: set of flows in the network.
- CP_f: set of candidate paths that obey the constraint 5 to the flow f.

Input : n, F, E, CP, a_{sd}^{f}
Output: F, a_{sd}^{f}, P_n

1 $B_{prev} \leftarrow \max\limits_{e_{sd} \in E} \left\{ \sum_{f \in F} a_{sd}^{f} \right\}$
2 $MBP_n \leftarrow arg \min\limits_{P \in CP_n} \max\limits_{e_{sd} \in P} \left\{ \sum_{f \in F} a_{sd}^{f} \right\}$
3 **Select** $Q \in MBP_n$ such that $|Q|$ is minimum
4 $a_{sd}^{n} \leftarrow 1 \ \ , \forall e_{sd} \in Q$
5 $P_n \leftarrow Q$
6 $B_{cur} \leftarrow \max\limits_{e_{sd} \in E} \left\{ \sum_{f \in F} a_{sd}^{f} \right\}$
7 **if** $B_{cur} > B_{prev}$ **then**
8   ReduceBottleneck(F, E, CP, a_{sd}^{f}, B_{cur})
9 **end**

Algorithm 1: Add route to new flow

Algorithm 1 describes how BPR operates when a new flow arrives in the network. This algorithm is based on the following stages: computing the network bottleneck before adding the new flow n (line 1); computing the set of feasible paths that enable it to follow route P_n with the smallest bottleneck increment without changing the routes of the other flows in the network (line 2); selecting the shortest path in this set of paths (line 3); routing the new flow along the selected path (lines 4 and 5); computing the network bottleneck after adding n (line 6); if the bottleneck has increased, trying to re-route some other flow, which was passing through the bottleneck before n was added, and thus seeking to reduce the bottleneck to the previous value or to reduce the path length of the other flows (lines 7–9).

The ReduceBottleneck algorithm, which will be described in detail later, seeks to re-route flows away from the bottleneck edges in order to return the bottleneck to the previous value. As can be noticed in lines 2 and 3 of Algorithm 1, the path selected for the flow n is a shortest path, as well as reducing the network bottleneck. Hence, the only opportunity to improve the result involves attempting to re-route any flow that was already passing through the bottleneck before the addition of n.

Algorithm 2 describes the actions taken by BPR whenever a flow has finished. This algorithm is based on the following stages: computing the network bottleneck (line 1); checking for edges in the path of the finished

Input : t, P_t, F, E, CP, a_{sd}^f
Output: F, a_{sd}^f
1 $B_{prev} \leftarrow \max\limits_{e_{sd} \in E} \left\{ \sum_{f \in F} a_{sd}^f \right\}$
2 $E_{AUX} \leftarrow \left\{ e_{sd} \in P_t \mid \sum_{f \in F} a_{sd}^f = B_{prev} - 1 \right\}$
3 $a_{sd}^t \leftarrow 0 \ , \forall e_{sd} \in P_t$
4 $F \leftarrow F - t$
5 $B_{cur} \leftarrow \max\limits_{e_{sd} \in E} \left\{ \sum_{f \in F} a_{sd}^f \right\}$
6 **if** $B_{cur} = B_{prev}$ **and** $|E_{AUX}| > 0$ **then**
7   ReduceBottleneck(F, E, CP, a_{sd}^f, B_{cur})
8 **end**

Algorithm 2: Remove finished flow

flow t that could eventually reduce the network bottleneck (line 2); effectively deleting the information of the finished flow (lines 3 and 4); computing the network bottleneck after removing flow t (line 5); seeking to reduce the bottleneck or the path length of other flows (if there was no reduction in the bottleneck through the deletion of the finished flow t and if there is an edge that offers this opportunity) (lines 6–8).

Naturally, when a flow has been completed, the network bottleneck can be reduced, and in this event the ReduceBottleneck algorithm is not run. However, if the bottleneck value remains unchanged after the finished flow has been removed, it may be possible to achieve an improvement by exploiting the room left in one of the edges of this finished flow. There is an opportunity to reduce the bottleneck if any edge of the finished flow has a number of flows equal to $B_{prev} - 1$. This means that one of the flows in the bottleneck can potentially use this opportunity, which is determined by the ReduceBottleneck algorithm.

Algorithm 3 is employed to carry out the ReduceBottleneck procedure, which is based on the following stages: calculating the set of associated links with the network bottleneck (line 1); checking each flow which passes through the network bottleneck (line 2), checking whether there is another path that offers a smaller bottleneck for the flow or has the same bottleneck but reduces the number of hops (lines 3–6); changing the route of this flow if the condition is right (lines 7–9); recalculating the network bottleneck (line 10) and the set of associated links with this bottleneck value (line 11). If there is a single

Input : F, E, CP, a_{sd}^{f}, B_{cur}
Output: a_{sd}^{f}

$E_B \leftarrow \left\{ e_{sd} \in E \mid \sum_{f \in F} a_{sd}^{f} = B_{cur} \right\}$
for *each* $\hat{f} \in F$ *such that* $a_{sd}^{\hat{f}} = 1$ *with* $e_{sd} \in E_B$ **do**
  $B_{aux} \leftarrow \min\limits_{P \in CP_{\hat{f}}} \left\{ \max\limits_{e_{sd} \in P} \left\{ \sum_{f \in F} a_{sd}^{f} \right\} \right\}$
  $MBP_{\hat{f}} \leftarrow arg \min\limits_{P \in CP_{\hat{f}}} \left\{ \max\limits_{e_{sd} \in P} \left\{ \sum_{f \in F} a_{sd}^{f} \right\} \right\}$
  Select $Q \in MBP_{\hat{f}}$ such that $|Q|$ is minimum
  if $B_{aux} + 1 < B_{cur}$ **or** $(B_{aux} + 1 = B_{cur}$ **and** $|Q| < |P_{\hat{f}}|)$ **then**
    $a_{sd}^{\hat{f}} \leftarrow 0 \ \forall e_{sd} \in P_{\hat{f}}$
    $a_{sd}^{\hat{f}} \leftarrow 1 \ \forall e_{sd} \in Q$
    $P_{\hat{f}} \leftarrow Q$
    $B_{cur} \leftarrow \max\limits_{e_{sd} \in E} \left\{ \sum_{f \in F} a_{sd}^{f} \right\}$
    $E_B \leftarrow \left\{ e_{sd} \in E \mid \sum_{f \in F} a_{sd}^{f} = B_{cur} \right\}$
  end
end

Algorithm 3: ReduceBottleneck

edge in the network bottleneck and the first condition in line 6 applies, then re-routing flow $\hat{f}$ will reduce the network bottleneck.

The execution of ReduceBottleneck whenever a flow is created or finished helps keep the network bottleneck under strict control. However, under very dynamic conditions that involve a large number of flows arriving and leaving the network in short time windows, BPR could only be adapted to run the ReduceBottleneck algorithm after a certain time interval. In such very dynamic scenarios, running the ReduceBottleneck when it is based on a time interval can decrease the computational cost at the expense of a wider fluctuation in the network bottleneck.

Since we chose to design BPR as a centralized scheme, we present some considerations about this choice in comparison with a distributed approach. In general, distributed routing approaches are useful when devices are handled independently or different parts of the network are able to operate autonomously. However, these characteristics do not apply to LWB networks. For example, in the Internet, where ASs (Autonomous Systems) are operated by independent entities, a distributed routing approach is mandatory. On the other hand, an LWB network is operated by a company or a consortium, which makes both distributed and centralized approaches viable. If a network is partitioned (e.g., due to link failures), a distributed routing solution allows multiple network partitions to operate independently. However, the clients of an LWB network would not benefit from this since they are only interested in having access to the Internet.

The centralized approach has several advantages for LWB networks. In a centralized routing approach, only one entity, e.g., a network controller, needs to keep information about the whole topology, which reduces the control of traffic and requires no time for algorithm convergence. The wireless routers then have less need for processing or memory, and can thus be cheaper and consume less energy. Since the routes are distributed from a central entity, it is also easier to handle security issues.

5. Performance Evaluation

As we previously described, JRCAR [24] is a relevant solution for the joint routing and channel assignment approach, but it is also a solution concerned with the same properties as JILP. Our proposal employs a channel assignment solution similar to JRCAR, and thus a performance evaluation should be concerned with the routing heuristics for

BPR and LBR. However, the choices made in each routing heuristic have an effect on the whole joint approach. Thus, we decided to evaluate and compare the joint approaches of JILP and JRCAR. As stated earlier, JRCAR was chosen because it is a state of the art solution that has been proven to achieve a high performance. In the following subsections we will describe the wide range of tests carried out in the ns-3 (Network Simulator 3) [56]; these employed different configurations and estimated several performance metrics.

5.1. Simulator Settings

The simulated scenarios were based on two types of topologies. Topology A has 45 network nodes that are randomly distributed in an area of 700 m $\times$ 700 m, and Topology B has 80 network nodes spread over an area of 1000 m $\times$ 1000 m. All the network nodes are fixed and have routing capability. There is only one external/Internet gateway, which is placed in the center of the network. This gateway is the border between the LWB and the other, external, networks. The positions of the nodes are randomly selected, but they have to comply with a number of restrictions: all the topologies must be connected graphs, the minimum distance between the network nodes must be 100 m, and the connectivity between the nodes must be defined by a physical layer model. We employed the Yans [57] as this physical layer model, using a parametrization that establishes connectivity between the nodes within a range of 150 m. We adjusted the transmission power and the receiver sensitivity so that the SNR could be kept high enough to obtain the same channel capacity within range of 150 m. Naturally, interference can reduce the effective throughput.

We investigated the impact of the number of flows on the network, and the impact of the number of available orthogonal channels on the channel assignment. Initially, the number of flows ranged from 10 to 50, while the number of radios in each router remained fixed at 5, and the number of orthogonal channels available for assignment was fixed at 12. After this, the number of orthogonal channels varied from 4 to 12, while the number of radios in each router remained fixed at 5, and the number of flows at 30.

We generated 60 random scenarios for every combination of topology and set of evaluated parameters. The traffic load consisted of TCP flows that had the gateway as their source, and the routers were randomly selected as destinations. A single router can be the destination of several flows, and thus the flows are identified by their source and destination transport ports. The flows start randomly at a time interval between 30.0 and 32.5 seconds. Each flow transfers 512 KB of data and the size of each packet is 2048 bytes. A simulation was only completed after all the flows had completed transferring all their data. The mean values are shown with confidence interval bounds at a confidence level of 95%. Table 2 provides a summary of the parameters for the simulations.

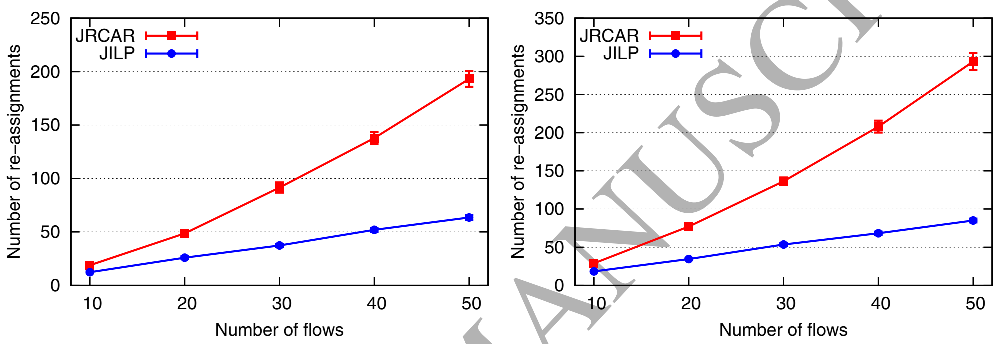
Table 2: Simulation parameters.

Parameter	Value
Transmission rate	11 Mbps (fixed)
Propagation model	Log-distance, exponent = 2.7
RTS/CTS	disabled
Transmission power	22 dBm
Energy detection threshold (EDth)	-96 dBm (ns-3 default)
Carrier sense threshold (CSth)	-99 dBm (ns-3 default)
TCP congestion control	NewReno (ns-3 default)

5.2. Channel Re-assignments

Channel re-assignments implies on temporary disconnections in the wireless links that reduce the application performance. Additionally, channel reassignment must follow a judicious strategy to avoid a cascade effect that can affect the whole network, and increase the signaling overhead [58]. Thus, it is important to keep control of the number of channel re-assignments by smoothing out frequent re-assignments.

The dynamics of the network flows, i.e., the fluctuation in the number of flows, has a significant impact on the network traffic pattern. Another important factor in the network traffic pattern is how the routing and channel assignment algorithms deal with the dynamics of the network flows. The routing algorithms based on JILP seek to make adjustments in the routes of these flows by exploiting opportunities for reducing the bottleneck and/or path length. The channel assignment algorithms of both joint approaches (JILP and JRCAR) seek to match the channels according to their interference and the links according to their loads. Naturally, the goal is to assign the channels with the lowest level of interference to the links with the highest load. Figure 1 shows the impact of both joint approaches on the number of channel re-assignments as a function of the number of flows in the network. As the number of flows increases, the number of channel re-assignments also increases, since the channel assignment algorithm seeks to reduce the levels of interference and contention in the network, particularly in the overloaded links. As expected, Topology B has a higher number of channel re-assignments than Topology A, due to the difference in the number of nodes, which means there is a difference in the levels of interference and contention.



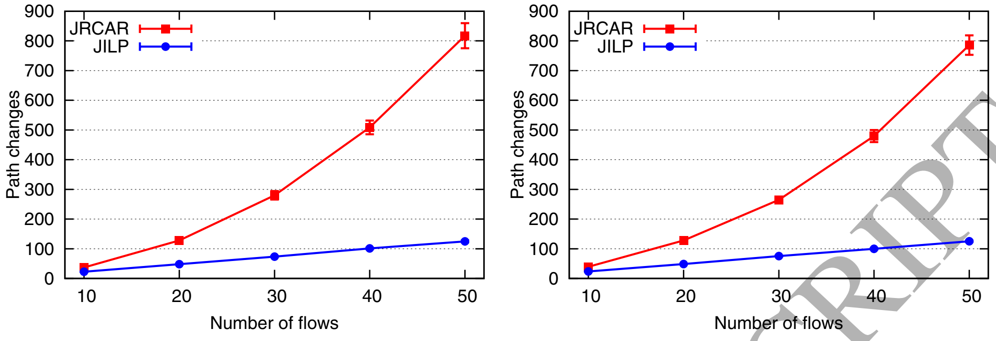
(a) Topology A

(b) Topology B

Figure 1: Number of channel re-assignments as a function of the number of flows.

In the presence of 20 flows or more, our proposal (JILP) has a much smaller number of channel re-assignments than JRCAR, thanks to the BPR algorithms. The efficiency of BPR can be confirmed by analyzing the traditional routing flapping (or path changes) metric, as illustrated in Figure 2. BPR is an on-demand routing heuristic that makes localized changes, while LBR tends to proactively reconfigure the routes of most of the flows. Since these changes have a harmful effect on the overall performance, the LBR strategy does not bring any benefits to the other metrics in the dynamics of a traditional network, as will be shown in the following subsections.

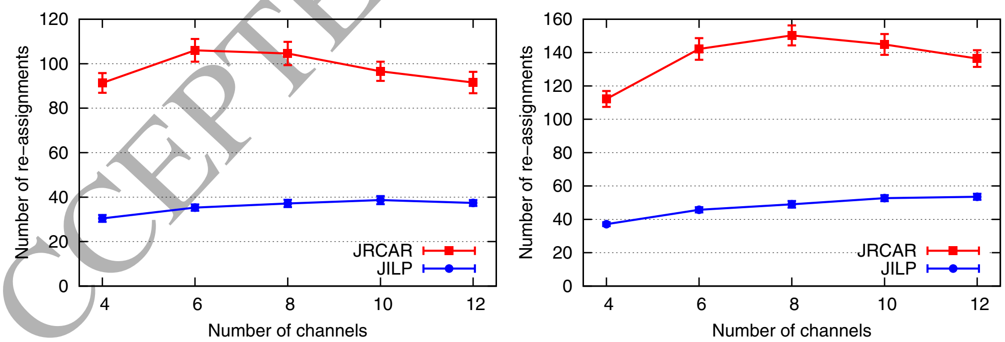
Figure 3 shows how the number of orthogonal channels affects the number of channel re-assignments. An increase in the number of orthogonal channels may imply that there is an increasing number of channel re-assignments if there are opportunities for less interference or contention. However, after a certain value, an increase in the number of orthogonal channels implies a reduction in the number of channel re-assignments since the algorithms have less work to do, i.e., there are fewer opportunities to lessen the interference



(a) Topology A (b) Topology B

Figure 2: Number of path changes as a function of the number of flows.

or contention. Naturally, the inflection point in the number of channel re-assignments is affected by other factors, such as the routing algorithms and the number of nodes: for example, Topology A versus Topology B. As expected, JILP maintains the number of channel re-assignments to be notably smaller than those of JRCAR. Additionally, JILP moves the inflection point ahead, which means that our proposal is more efficient in using the available orthogonal channels. The reason for this is that BPR creates shorter average paths than LBR, which suggests there is a smaller number of active links and also less interference and contention.



(a) Topology A (b) Topology B

Figure 3: Number of channel re-assignments as a function of the number of channels.

5.3. Aggregate Throughput

In this part of the performance evaluation, there is an analysis of the average aggregate throughput of the network flows, which is described as follows. To start with, the average throughput of each network flow is calculated on the basis of the following expression:

$$\rho_f = \frac{b_f}{T_f - B_f}, \tag{7}$$

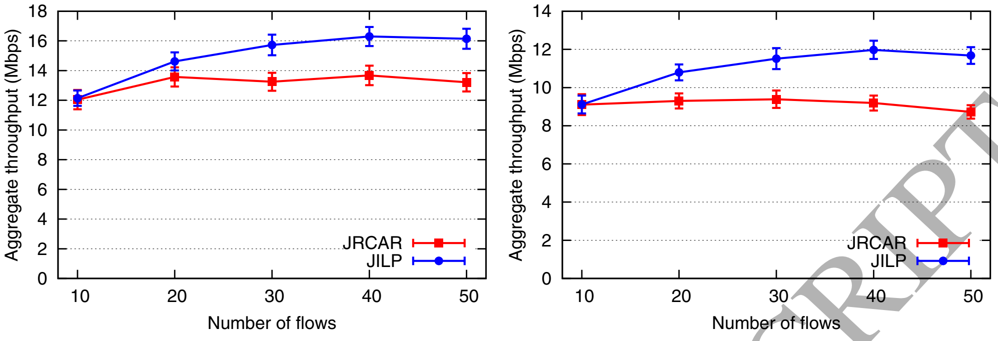
where ρ_f is the average throughput of the flow $f \in F$, b_f is the amount of bits transferred by the flow, T_f is when the flow f finished, and B_f is when the flow started. Next, the average aggregate throughput of the flows is calculated for all the scenarios according to the following expression:

$$\rho_{aa} = \frac{1}{|S|} \sum_{s \in S} \sum_{f \in F} \rho_f, \tag{8}$$

where ρ_{aa} is the average aggregate throughput of the network flows, and ρ_f was defined by Equation 7 for a specific scenario $s \in S$.

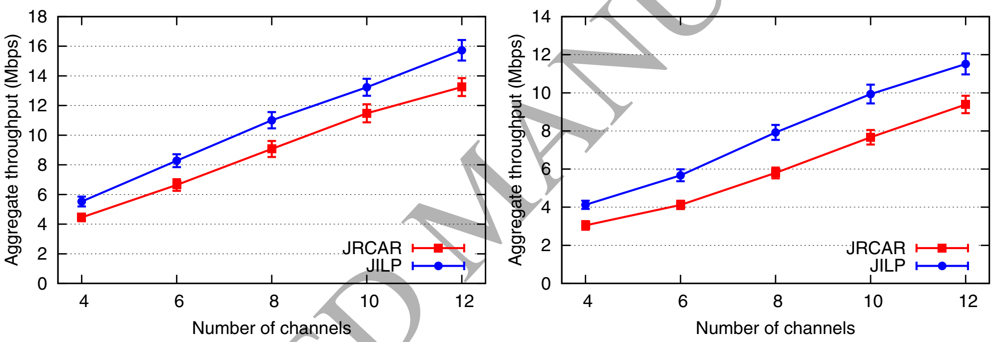
Figure 4 shows the average aggregate throughput as a function of the number of flows in the network. As expected, the aggregate throughput increases as the number of flows increases, until the network is fully saturated, and then the aggregate throughput starts to decline. Again, the number of network nodes and the routing algorithms affect the inflection point. Owing to the shorter paths chosen by BPR and the resulting lesser interference and contention achieved by JILP, our proposal has a distinct advantage compared with that of JRCAR. Naturally, this advantage tends to increase as the network load (i.e., the number of flows) and the number of opportunities (i.e., the number of nodes and links) increases. For example, JILP provides an average aggregate throughput close to 33% higher than JRCAR in Topology B when there are 50 flows in the network.

In Figure 5 it can be confirmed that an increase in the orthogonal channels implies an increase in the aggregate throughput. This was expected since more orthogonal channels means less interference and contention. JILP outperforms JRCAR consistently in all the scenarios, as occurred in the previous results. As in Figure 4, in Figure 5 it can also be observed that Topology A has an average aggregate throughput that is higher than that of Topology B. This is because Topology B requires longer path lengths, since there are



(a) Topology A (b) Topology B

Figure 4: Average aggregate throughput as a function of the number of flows.



(a) Topology A (b) Topology B

Figure 5: Average aggregate throughput as a function of the number of channels.

more nodes and links. Having longer paths increases the end-to-end delay and hence reduces the transmission rate of the TCP flows.

In our previous paper [26], we estimated some performance metrics such as the bottleneck value, path length, and route updating, which suggested BPR had potential benefits. In this article, by adopting the joint approach JILP, we have confirmed our initial findings and have been able to quantify the improvement in throughput in important LWB scenarios. In the following subsection, we point out some of the additional advantages of our proposal.

5.4. Fairness and Minimum Throughput

The fairness measures between the network flows and the minimum throughput achieved by any flow are important metrics because they may reveal undesired factors that do not appear in metrics such as the aggregate throughput. Naturally, it is not desirable to obtain improved aggregate throughput at the cost of flow starvation or severe unfairness, for example. If this situation arises, it means that although the overall performance seems better, some users may receive a very poor service or no service at all. In general, fairness is the main metric used to investigate this sort of issue. However, we have observed in our data that a metric of fairness was not able to properly quantify the inequalities, which is what has led to the use of the metric for average minimum throughput.

When estimating fairness, we employed the fairness index designed by Jain [59], which is based on the following expression:

$$\theta = \frac{(\sum_{f \in F} \rho_f)^2}{n * \sum_{f \in F} (\rho_f)^2}, \tag{9}$$

where θ represents the fairness index between the network flows in a specific simulation scenario $s \in S$, ρ_f is the average throughput of each flow $f \in F$, and n is the number of network flows. Thus, the average fairness between all scenarios of a certain parametrization is trivially obtained:

$$F_a = \frac{1}{|S|} \sum_{s \in S} \theta. \tag{10}$$

In calculating the average minimum throughput, we initially estimated the smaller throughput obtained in the network, that is: $T_m = \min \rho_f, \forall f \in F$. Then the average minimum throughput can also be trivially obtained:

$$T_{ma} = \frac{1}{|S|} \sum_{s \in S} T_m. \tag{11}$$

In this part of the evaluation, we will focus on Topology B because the results have a similar trend as in Topology A, as has been shown in the previous subsections. Figures 6 and 7 display the average of Jain's fairness index and the average minimum throughput as a function of the number of flows. As the number of flows increases, the variation in the path lengths also increases, i.e., there is an increase in the variance of the end-to-end delay and hence in the variance of the throughput. Thus, there is a natural decline in the average fairness index. The decrease in average minimum throughput was expected since the number of flows sharing common network resources increases. Both solutions, JILP and JRCAR, can be employed to mitigate this issue, but the network bottleneck limits the extent of the improvement. The difference between JILP and JRCAR is not clear for most values in the average fairness index, although JILP is always superior. However, JILP is able to keep an average minimum throughput that is notably higher than JRCAR, especially when there is a heavy load. For example, when there are 50 flows in the network, JILP provides an average minimum throughput of 96 Kbps while JRCAR offers 60 Kbps.

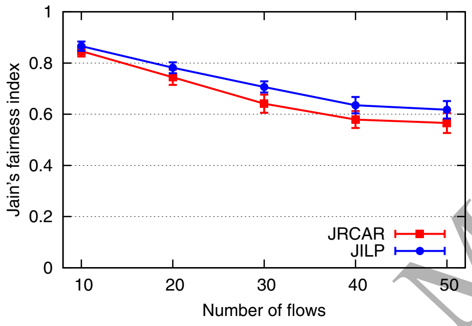


Figure 6: Fairness as a function of the number of flows (Topology B).

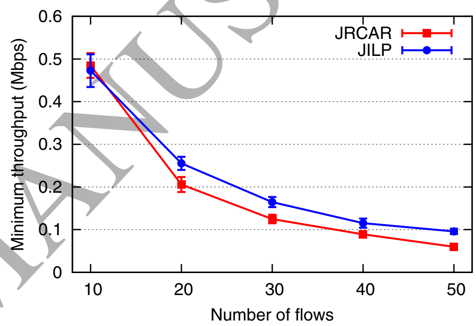


Figure 7: Minimum throughput as a function of the number of flows (Topology B).

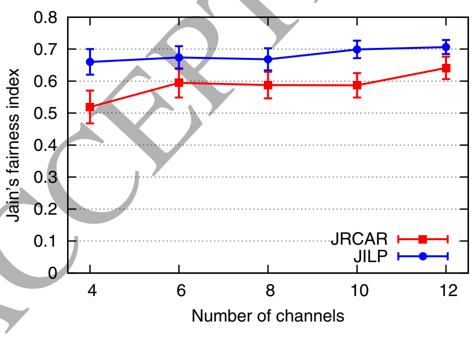


Figure 8: Fairness as a function of the number of channels (Topology B).

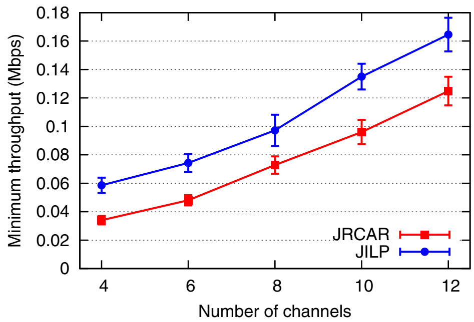


Figure 9: Minimum throughput as a function of the number of channels (Topology B).

Figures 8 and 9 show the average fairness index and the average minimum throughput as functions of the number of orthogonal channels. Since an increase in the number of channels means an increase in the amount of shared resources, there is an increase in the average fairness index and in the average minimum throughput. While JILP always maintains a higher average fairness index than JRCAR, the latter seems to be more sensitive to an increase in the number of channels. The reason for this is that LBR generates longer path lengths than BPR, which creates a higher demand for the orthogonal channels required to reduce contention and interference. As a result, the channel assignment algorithms tend to improve the average fairness index while seeking to alleviate the links, in particular, those that are most congested. Despite the similarity in the average fairness index when there are 12 channels (Figure 8), there is a noticeable advantage in using JILP in terms of the average minimum throughput (Figure 9). Again, the main reason is the routing algorithms employed by JILP, which, on average, create shorter path lengths and have an improved stability.

5.5. Additional Experiments

This section presents an additional performance evaluation of JILP under different traffic flows and link rates. This evaluation employed Topology A, i.e., 45 network nodes, and the five sets of traffic mix listed in Table 3. Each set is composed of approximately 60% TCP, 30% low-rate UDP (100 Kbps), and 10% high-rate UDP (1.5 Mbps) [60]. Low-rate UDP represents a voice communication flow, e.g., a Skype conversation. High-rate UDP represents a video flow, e.g., a Netflix video stream. Additionally, the transmission rate is automatically chosen by the Minstrel rate control algorithm [61], which is used in many real-world devices. Thus, the rate of each link varies over time according to the channel conditions. The rest of the configuration follows the description presented in Section 5.1.

Figure 10 illustrates how the number of flows affects the number of channel re-assignments, under this different scenario of traffic and link rates. As in the previous results, the number of channel re-assignments increases as the number of flows increases, and JILP notably outperforms JRCAR. This confirms the efficiency of our heuristic BPR, which is activated on-demand and makes localized changes.

In Figure 11, we present the average loss rate observed in the UDP traffic. Even with the loss rate values' being low, they may have an impact on time sensitive applications. The stabler routes and shorter paths, which affect the

Table 3: Sets of traffic mix.

Traffic mix	TCP	Low-rate UDP	High-rate UDP	Aggregate
A	9	4	2	15
B	18	9	3	30
C	27	13	5	45
D	36	18	6	60
E	45	22	8	75

contention, bear the main responsibility for making JILP significantly better than JRCAR in terms of this metric. The UDP loss rate is also an indication of the network condition experienced by the aggregate traffic mix. Thus, we can expect that the TCP flows experience a lower loss rate (or higher throughput) when BPR is the routing solution.

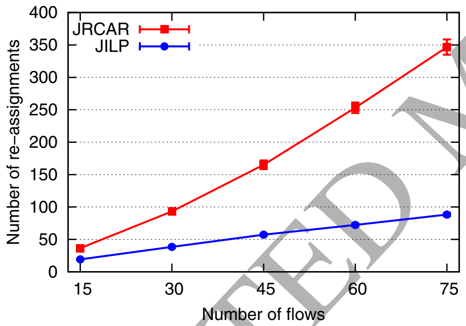


Figure 10: Number of channel re-assignments as a function of the number of flows (Topology A).

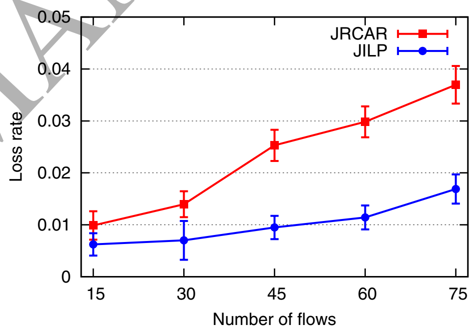


Figure 11: Average loss rate as a function of the number of flows (Topology A).

Figure 12 shows the average throughput of the five sets of traffic mix listed in Table 3. As expected, JILP performs better than JRCAR for TCP traffic. This is consistent with the results presented in the previous sections. JILP also provides some benefits to the high-rate UDP, mainly when the network load is high. The loss rate presented in Figure 11 already suggested this result.

Naturally, this environment, with dynamic rate adaptation and different types of traffic, has an impact on the performance of JILP. For example,

Figure 13 presents the minimum throughput as a function of the number of flows. The highest minimum throughput is close to 100 Kbps because this is the value for the low-rate UDP traffic. JILP outperforms JRCAR only when the network load is high, and the advantage is less noticeable than the previous results.

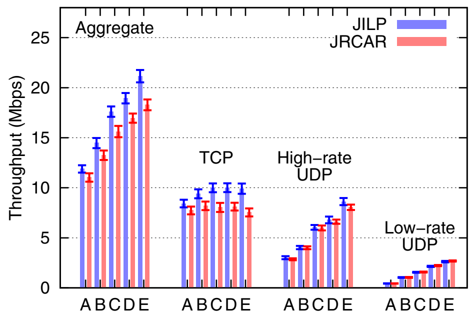


Figure 12: Throughput of the different sets of traffic mix (Topology A).

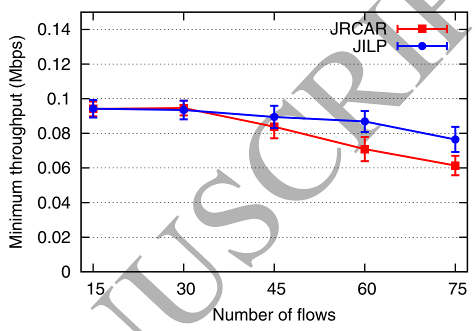


Figure 13: Minimum throughput as a function of the number of flows (Topology A).

In summary, both JILP and JRCAR are affected by the different types of traffic and the rate adaptation algorithm. However, the performance of JILP is superior to the JRCAR in most of the metrics and scenarios evaluated.

6. Conclusions and Future Research

In this article, we have presented a joint solution involving a routing and channel assignment for wireless mesh networks in the context of low-cost wireless backhauls. The core of our proposal, called JILP, is a set of routing algorithms designed for bottleneck reduction and path length control. We have evaluated and compared our solution with a state of the art approach [24], called JRCAR, using the network simulator ns-3. The results show that JILP outperforms JRCAR in terms of evaluated metrics, including aggregate throughput and fairness. A fuller assessment of our proposal and an illustration of its key features have been provided by evaluating complementary metrics such as the number of channel re-assignments and the number of path changes. In the performance evaluation, we took into account the different load conditions (number of flows in the network) and different amount of resources (number of orthogonal channels).

In addition, we also analyzed the impact of routing heuristics on the joint approach adopted to improve the traffic performance, where both joint approaches use the same channel assignment algorithm. It was found that the routing heuristic has a non-negligible impact on traffic performance. JILP employs our BPR routing heuristic, while JRCAR uses the LBR routing heuristic. BPR offers an on-demand solution that employs two algorithms based on local routing decision-making for every created or finished flow. Thus, BPR provides a more efficient trade-off between the network bottleneck and the reduction of the path length in an efficient way, which can combine these factors in a single solution. In future research, we intend to implement BPR as a fully-functional routing protocol for real-world networks.

Acknowledgments

This work has been partially supported by the grant 204486/2014-9 from the Brazilian Research Agency (CNPq).

References

[1] P. Agyapong, M. Iwamura, D. Staehle, W. Kiess, A. Benjebbour, Design considerations for a 5G network architecture, Communications Magazine, IEEE 52 (11) (2014) 65–75.

[2] C.-X. Wang, F. Haider, X. Gao, X.-H. You, Y. Yang, D. Yuan, H. Aggoune, H. Haas, S. Fletcher, E. Hepsaydir, Cellular architecture and key technologies for 5G wireless communication networks, Communications Magazine, IEEE 52 (2) (2014) 122–130.

[3] E. Hossain, M. Rasti, H. Tabassum, A. Abdelnasser, Evolution toward 5G multi-tier cellular wireless networks: An interference management perspective, Wireless Communications, IEEE 21 (3) (2014) 118–127.

[4] A. Osseiran, F. Boccardi, V. Braun, K. Kusume, P. Marsch, M. Maternia, O. Queseth, M. Schellmann, H. Schotten, H. Taoka, H. Tullberg, M. Uusitalo, B. Timus, M. Fallgren, Scenarios for 5G mobile and wireless communications: the vision of the METIS project, Communications Magazine, IEEE 52 (5) (2014) 26–35.

[5] L. Doyle, J. Kibilda, T. Forde, L. DaSilva, Spectrum Without Bounds, Networks Without Borders, Proceedings of the IEEE 102 (3) (2014) 351–365.

[6] P. Demestichas, A. Georgakopoulos, D. Karvounas, K. Tsagkaris, V. Stavroulaki, J. Lu, C. Xiong, J. Yao, 5G on the Horizon: Key Challenges for the Radio-Access Network, Vehicular Technology Magazine, IEEE 8 (3) (2013) 47–53.

[7] S. Andreev, A. Pyattaev, K. Johnsson, O. Galinina, Y. Koucheryavy, Cellular traffic offloading onto network-assisted device-to-device connections, Communications Magazine, IEEE 52 (4) (2014) 20–31.

[8] N. Bhushan, J. Li, D. Malladi, R. Gilmore, D. Brenner, A. Damnjanovic, R. Sukhavasi, C. Patel, S. Geirhofer, Network densification: the dominant theme for wireless evolution into 5G, Communications Magazine, IEEE 52 (2) (2014) 82–89.

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

[10] J. Bicket, D. Aguayo, S. Biswas, R. Morris, Architecture and Evaluation of an Unplanned 802.11B Mesh Network, in: Proceedings of the 11th Annual International Conference on Mobile Computing and Networking, MobiCom '05, 2005, pp. 31–42.

[11] J. Ishmael, S. Bury, D. Pezaros, N. Race, Deploying Rural Community Wireless Mesh Networks, Internet Computing, IEEE 12 (4) (2008) 22–29.

[12] R. Draves, J. Padhye, B. Zill, Routing in Multi-radio, Multi-hop Wireless Mesh Networks, in: Proceedings of the 10th Annual International Conference on Mobile Computing and Networking, 2004, pp. 114–128.

[13] B. Milic, M. Malek, Properties of Wireless Multihop Networks in Theory and Practice, in: Guide to Wireless Ad Hoc Networks, Computer Communications and Networks, 2009, pp. 1–26.

[14] A. Raniwala, T. cker Chiueh, Architecture and algorithms for an IEEE 802.11-based multi-channel wireless mesh network, in: INFOCOM 2005.

24th Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings IEEE, Vol. 3, 2005, pp. 2223–2234 vol. 3.

[15] P. Gupta, P. Kumar, The capacity of wireless networks, Information Theory, IEEE Transactions on 46 (2) (2000) 388–404.

[16] M. Marina, S. Das, A topology control approach for utilizing multiple channels in multi-radio wireless mesh networks, in: Broadband Networks, 2005. BroadNets 2005. 2nd International Conference on, 2005, pp. 381–390.

[17] A. Subramanian, H. Gupta, S. Das, Minimum interference channel assignment in multi-radio wireless mesh networks, in: Sensor, Mesh and Ad Hoc Communications and Networks, 2007. SECON '07. 4th Annual IEEE Communications Society Conference on, 2007, pp. 481–490.

[18] L. Ma, M. Denko, A Routing Metric for Load-Balancing in Wireless Mesh Networks, in: Advanced Information Networking and Applications Workshops, 2007, AINAW '07. 21st International Conference on, 2007, pp. 409–414.

[19] D. Manikantan Shila, T. Anjali, Load Aware Traffic Engineering for Mesh Networks, Comput. Commun. 31 (7) (2008) 1460–1469.

[20] L. T. Nguyen, R. Beuran, Y. Shinoda, A Load-aware Routing Metric for Wireless Mesh Networks, in: Computers and Communications, 2008. ISCC 2008. IEEE Symposium on, 2008, pp. 429–435.

[21] A. Raniwala, K. Gopalan, T.-c. Chiueh, Centralized Channel Assignment and Routing Algorithms for Multi-channel Wireless Mesh Networks, SIGMOBILE Mob. Comput. Commun. Rev. 8 (2) (2004) 50–65.

[22] J. Tang, G. Xue, W. Zhang, Interference-aware Topology Control and QoS Routing in Multi-channel Wireless Mesh Networks, in: Proceedings of the 6th ACM International Symposium on Mobile Ad Hoc Networking and Computing, 2005, pp. 68–77.

[23] M. Alicherry, R. Bhatia, L. E. Li, Joint Channel Assignment and Routing for Throughput Optimization in Multiradio Wireless Mesh Networks, IEEE Journal on Selected Areas in Communications 24 (11) (2006) 1960–1971.

[24] J. J. Gálvez, P. M. Ruiz, Efficient Rate Allocation, Routing and Channel Assignment in Wireless Mesh Networks Supporting Dynamic Traffic Flows, Ad Hoc Networks 11 (6) (2013) 1765–1781.

[25] D. Wu, S.-H. Yang, L. Bao, C. H. Liu, Joint Multi-radio Multi-channel Assignment, Scheduling, and Routing in Wireless Mesh Networks, Journal of Wireless Networks (Springer) 20 (1) (2014) 11–24.

[26] M. O. M. C. De Mello, V. C. M. Borges, L. D. L. Pinto, K. V. Cardoso, Load balancing routing for path length and overhead controlling in wireless mesh networks, in: Computers and Communication (ISCC), 2014 IEEE Symposium on, 2014, pp. 1–6.

[27] Y. Bejerano, S.-J. Han, A. Kumar, Efficient Load-balancing Routing for Wireless Mesh Networks, Computer Networks 51 (10) (2007) 2450–2466.

[28] P.-H. Hsiao, A. Hwang, H. Kung, D. Vlah, Load-balancing Routing for Wireless Access Networks, in: INFOCOM 2001. Twentieth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE, 2001, pp. 986–995.

[29] H. Dai, R. Han, A Node-centric Load Balancing Algorithm for Wireless Sensor Networks, in: Global Telecommunications Conference, 2003. GLOBECOM '03. IEEE, 2003, pp. 548–552.

[30] H.-G. Choi, S.-J. Han, Domain Load Balancing Routing for Multi-gateway Wireless Mesh Networks, Journal of Wireless Networks. Springer Kluwer Academic Publishers 16 (8) (2010) 2105–2122.

[31] V. C. M. Borges, M. Curado, E. Monteiro, Cross-Layer Routing Metrics for Mesh Networks: Current Status and Research Directions, Computer Communications 34 (6) (2011) 681–703.

[32] L. Zhao, A. Y. Al-Dubai, G. Min, An efficient Neighbourhood Load Routing Metric for Wireless Mesh Networks, Simulation Modelling Practice and Theory 19 (6) (2011) 1415–1426.

[33] Z. Hao, Y. Li, An adaptive load-aware routing algorithm for multi-interface wireless mesh networks, Wireless Networks 21 (2) (2015) 557–564.

[34] K. N. Ramachandran, I. Sheriff, E. M. Belding, K. C. Almeroth, Routing Stability in Static Wireless Mesh Networks, in: Proceedings of the 8th International Conference on Passive and Active Network Measurement, 2007, pp. 73–83.

[35] D. Nandiraju, L. Santhanam, N. Nandiraju, D. Agrawal, Achieving load balancing in wireless mesh networks through multiple gateways, in: Mobile Adhoc and Sensor Systems (MASS), 2006 IEEE International Conference on, 2006, pp. 807–812.

[36] B. Radunovic, C. Gkantsidis, P. Key, P. Rodriguez, An optimization framework for opportunistic multipath routing in wireless mesh networks, in: INFOCOM 2008. The 27th Conference on Computer Communications. IEEE, 2008, pp. –.

[37] W. Si, S. Selvakennedy, A. Y. Zomaya, An overview of channel assignment methods for multi-radio multi-channel wireless mesh networks, Journal of Parallel and Distributed Computing 70 (5) (2010) 505–524.

[38] K. Ramachandran, E. Belding, K. Almeroth, M. Buddhikot, Interference-Aware Channel Assignment in Multi-Radio Wireless Mesh Networks, in: INFOCOM 2006. 25th IEEE International Conference on Computer Communications. Proceedings, 2006, pp. 1–12.

[39] M. Kodialam, T. Nandagopal, Characterizing the capacity region in multi-radio multi-channel wireless mesh networks, in: Proceedings of the 11th Annual International Conference on Mobile Computing and Networking, 2005, pp. 73–87.

[40] A. Brzezinski, G. Zussman, E. Modiano, Enabling distributed throughput maximization in wireless mesh networks: A partitioning approach, in: Proceedings of the 12th Annual International Conference on Mobile Computing and Networking, 2006, pp. 26–37.

[41] S. Avallone, I. Akyildiz, A channel assignment algorithm for multi-radio wireless mesh networks, in: Computer Communications and Networks, 2007. ICCCN 2007. Proceedings of 16th International Conference on, 2007, pp. 1034–1039.

[42] A. Raniwala, T. cker Chiueh, Architecture and algorithms for an IEEE 802.11-based multi-channel wireless mesh network, in: INFOCOM 2005.

24th Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings IEEE, 2005, pp. 2223–2234.

[43] S. Das, H. Pucha, D. Koutsonikolas, Y. Hu, D. Peroulis, Dmesh: Incorporating practical directional antennas in multichannel wireless mesh networks, Selected Areas in Communications, IEEE Journal on 24 (11) (2006) 2028–2039.

[44] A. Naveed, S. Kanhere, S. Jha, Topology control and channel assignment in multi-radio multi-channel wireless mesh networks, in: Mobile Adhoc and Sensor Systems, 2007. MASS 2007. IEEE Internatonal Conference on, 2007, pp. 1–9.

[45] Ying Qu and Bryan Ng and Winston Seah, A Survey of Routing and Channel Assignment in Multi-channel Multi-radio WMNs, Journal of Network and Computer Applications 65 (2016) 120–130. doi:http://dx.doi.org/10.1016/j.jnca.2016.02.017.

[46] V. Gardellin, S. K. Das, L. Lenzini, C. Cicconetti, E. Mingozzi, G-PaMeLA: A divide-and-conquer Approach for Joint Channel Assignment and Routing in Multi-radio Multi-channel Wireless Mesh Networks, Journal of Parallel and Distributed Computing 71 (3) (2011) 381–396.

[47] S. M. Gammar, S. Ghannay, JRCAP: A Joint Routing and Channel Assignment Protocol for Multi-radio Multi-channel IEEE 802.11s Mesh Networks, Journal of Network and Systems Management 24 (1) (2015) 140–160.

[48] S. Avallone, G. D. Stasi, A New MPLS-Based Forwarding Paradigm for Multi-Radio Wireless Mesh Networks, IEEE Transactions on Wireless Communications 12 (8) (2013) 3968–3979. doi:10.1109/TWC.2013.071113.121529.

[49] S. Avallone, A. Banchs, A channel assignment and routing algorithm for energy harvesting multi-radio wireless mesh networks, IEEE Journal on Selected Areas in Communications PP (99) (2016) 1–1. doi:10.1109/JSAC.2016.2520238.

[50] J. Tang, G. Xue, W. Zhang, Maximum Throughput and Fair Bandwidth Allocation in Multi-Channel Wireless Mesh Networks, in: INFOCOM

2006. 25th IEEE International Conference on Computer Communications. Proceedings, 2006, pp. 1–10.

[51] S. Avallone, I. Akyildiz, G. Ventre, A Channel and Rate Assignment Algorithm and a Layer-2.5 Forwarding Paradigm for Multi-Radio Wireless Mesh Networks, Networking, IEEE/ACM Transactions on 17 (1) (2009) 267–280.

[52] J. J. G. García, Throughput optimization of Multi-hop Wireless Networks through interference-aware load balancing, Ph.D. thesis, University of Murcia (June 2011).

[53] J. Gao, L. Zhang, Trade-Offs between Stretch Factor and Load-Balancing Ratio in Routing on Growth-Restricted Graphs, IEEE Transactions on Parallel and Distributed Systems 20 (2) (2009) 171–179.

[54] B. He, B. Xie, D. P. Agrawal, Optimizing Deployment of Internet Gateway in Wireless Mesh Networks, Computer Communications 31 (7) (2008) 1259–1275.

[55] M. Ehrgott, X. Gandibleux, A survey and annotated bibliography of multiobjective combinatorial optimization, OR-Spektrum 22 (4) (2000) 425–460.

[56] G. Riley, T. Henderson, The ns-3 network simulator, in: Modeling and Tools for Network Simulation, 2010, pp. 15–34.

[57] M. Lacage, T. R. Henderson, Yet Another Network Simulator, in: Proceeding from the 2006 Workshop on Ns-2: The IP Network Simulator, 2006, pp. 1–10.

[58] N. Lilith, K. Dogancay, Reduced-State SARSA Featuring Extended Channel Reassignment for Dynamic Channel Allocation in Mobile Cellular Networks, in: Networking - ICN 2005, Vol. 3421 of Lecture Notes in Computer Science, 2005, pp. 531–542.

[59] R. Jain, The Art of Computer Systems Performance Analysis: Techniques for Experimental Design, Measurement, Simulation, and Modeling, Wiley-Interscience, 1991.

[60] V. Borges, E. Dimitrov, M. Curado, E. Monteiro, Performance assessment of cluster load balancing routing methods for triple play services in Wireless Mesh Networks, in: Network Operations and Management Symposium (NOMS), 2012 IEEE, 2012, pp. 974–980.

[61] D. Xia, J. Hart, Q. Fu, On the performance of rate control algorithm Minstrel, in: Personal Indoor and Mobile Radio Communications (PIMRC), 2012 IEEE 23rd International Symposium on, 2012, pp. 406–412.

Micael O. M. C. de Mello obtained a B.Sc. degree in Information Systems at Universidade Federal de Goiás (UFG), Goiânia, Brazil, in 2012, and an M.Sc. degree in Computer Science from the Universidade Federal de Goiás (UFG), Goiânia, Brazil, in 2014. His research interests include High-Speed Networks, Performance Evaluation, Wireless Networks, and Mobility.

Vinicius da Cunha Martins Borges: He obtained a BA degree in Computer science from the University of Rio Verde-Brazil (2003), a MSc degree from the Federal University of Santa Catarina-Brazil (2006) and a PhD from the University of Coimbra-Portugal (2013). He has taken part both as a reviewer and TPC member of various international conferences and journals (IEEE, ACM, Springer and Elsevier). Currently, he is Assistant Professor at the Universidade Federal de Goiás (Brazil). His main research interests are in Computer Networks, in particular, routing schemes for futuristic wireless networks.


Leizer Lima Pinto Adjunct Professor, Institute of Informatics, Universidade Federal de Goiás, Brazil. Researcher in Productivity at CNPq, (Level 2, Brazil). B.Sc. degree in Computer Science, Dept. of Computer Science, Pontifical Catholic University of Goiás, Brazil, 2004. M.Sc. degree in Systems Engineering and Computer Science, COPPE/UFRJ, Brazil, 2007. D.Sc., Systems Engineering and Computer Science, COPPE/UFRJ, Brazil, 2009. His research interests include Multi-objective Combinatorial Optimization and Network Optimization.


Kleber Vieira Cardoso is an Adjunct Professor at Universidade Federal de Goiás (UFG), Goiânia, Brazil, where he is the Head of the Computer Networks and Distributed Systems LABORAtory (LABORA). Prof. Kleber obtained a Ph.D. in Electrical Engineering from Federal University of Rio de Janeiro, Rio de Janeiro, 2009, the M.Sc. degree in Electrical Engineering from Federal University of Rio de Janeiro, Rio de Janeiro, 2002, and the B.Sc. in Computer Science from Universidade Federal de Goiás (UFG), Brazil, 1997. He is a member of TPC and has been on the Committee for several conferences and workshops in his fields of interest. He chaired the Graduate Program in Computer Science of UFG (2011-2013). His research interests include the Internet, High-Speed Networks, Performance Evaluation, Wireless Networks, Mobility, and Quality of Service. He is a member of the Brazilian Computer Society.