Ad Hoc Networks 29 (2015) 32–44

Contents lists available at ScienceDirect

Ad Hoc Networks

journal homepage: www.elsevier.com/locate/adhoc

Simulated annealing for interface-constrained channel assignment in wireless mesh networks

Ying-Yu Chen, Chien Chen *

Department of Computer Science, National Chiao Tung University, 1001 University Road, Hsinchu 300, Taiwan, ROC

ARTICLE INFO

Article history:
Received 19 May 2014
Received in revised form 16 January 2015
Accepted 30 January 2015
Available online 7 February 2015

Keywords:
Wireless mesh networks
Channel assignment
Interface constraint
Neighbor generation
Simulated annealing

ABSTRACT

The channel assignment problem in wireless mesh networks is considered in this paper. Due to the limited number of radio interfaces that each node is equipped with, channel assignment must ensure that the interface constraint is obeyed, i.e., the number of different channels assigned to the links incident on a node must not exceed the number of interfaces the node is equipped with. However, interface constraint may be one of the causes that hinder the performance of some of the existing heuristics. In this paper, we use simulated annealing to solve the channel assignment problem in wireless mesh networks. We consider using two neighbor generating approaches that handle the interface constraint in different ways. The first approach generates random solutions that need not conform to the interface constraint. With the incorporation of a penalty function technique, the solutions gradually converge to feasible solutions. The second approach allows only feasible solutions to be generated during the simulated annealing process. Simulations are conducted that compare our proposed simulated annealing algorithms with a tabu-based algorithm under different network settings and traffic scenarios. The results show that our algorithms exhibit improved network performance over the tabu-based algorithm.

1. Introduction

Wireless mesh networks (WMNs) [1,2] have drawn much attention due to their flexible, low-cost deployment and potential applications. WMNs are composed of two types of node, mesh routers and mesh clients. Mesh routers communicate with each other through wireless links and form a mesh backbone for the mesh clients. In a multi-hop fashion, packets in WMNs are delivered to their intended destinations by the mesh routers. Some of the mesh routers serve as gateways through which nodes within the WMN can get access to the wired Internet.

In WMNs, each mesh router can be equipped with multiple network interface cards (NICs), enabling it to utilize the radio spectrum more efficiently. With multiple channels, two close node pairs can communicate simultaneously without interfering with each other if they operate on different non-interfering channels. With multiple NICs, a node can further work on multiple channels simultaneously, which is not allowed with only one NIC. The availability of multiple channels and multiple interfaces poses a direct challenge of how to assign channels to the NICs to maximize performance gain [3–5].

There exist several variants of channel assignment problems in WMNs and many of them are NP-hard [6–8]. In one of its generic forms, given the topology (i.e., the position of mesh routers and the wireless links among them), expected load on the links, number of channels, number of interfaces of each node, and the interference

* Corresponding author. Tel.: +886 3 5712121x31768; fax: +886 3 5721490.

E-mail addresses: cyy125.cs96g@g2.nctu.edu.tw (Y.-Y. Chen), chienchen@cs.nctu.edu.tw (C. Chen).

http://dx.doi.org/10.1016/j.adhoc.2015.01.019
1570-8705/

model between nodes, the goal is to assign a proper channel to each of the given wireless links so as to optimize network performance, e.g., minimizing the interference and maximizing the throughput. In general, the number of NICs is limited, and thus one must ensure that the solution does not violate the interface constraint, which states that the number of different channels assigned to the links incident on a node must be no greater than the number of NICs the node is equipped with.

The necessity to conform to the interface constraint may sometimes restrain the performance of a seemingly promising heuristic for solving the channel assignment problem [9]. As a simple example for illustration, consider the WMN in Fig. 1(a) where each node has two NICs and there are four available channels. We are asked to assign a channel to each link. Assume a greedy-based approach has assigned links (A, B), (A, C), (B, D), and (C, D) in sequence to channels 1, 2, 3, and 4, respectively. Note that the four assigned links seem to be an optimal choice so far since they do not interfere with each other. However, link (B, C) cannot be assigned to any channel without breaking the interface constraint on either node B or node C. Suppose channel 1 is assigned to link (B, C), as shown in Fig. 1(b), then node C violates the interface constraint, which necessitates further modification. A feasible channel assignment may finally turn out to be like the one in Fig. 1(c), wherein the channel assigned to link (C, D) is changed from channel 4 to channel 1. In a large topology, this modification may involve more and more links until the interface constraint is finally resolved. This behavior of chain modifications to the already assigned links has been described previously [6,10] and is informally called the ripple effect.

To get a notion of the impact of the interface constraint on the performance of the heuristics to solve the channel assignment problem, we examine the tabu-based algorithm proposed by Subramanian et al. [7] for illustration. The algorithm consists of two phases. In phase one, the algorithm ignores the interface constraint and searches for a minimum network interference solution using a tabu-search technique. Then in phase two, violations of the interface constraint are progressively resolved by going through a series of *merge* operations (see Section 4 for details), which are prone to causing ripple effects, that repeatedly reduce the number of channels used by the nodes that violate the interface constraint. We implement the tabu-based algorithm, abbreviated by TABU, and a similar one with phase one being replaced by simulated annealing, abbreviated by SA. Fig. 2 shows the result conducted assuming a 50-node topology, 12 available channels, three NICs per node, and random single-hop traffic pattern. The legends with and without suffix -p1 represent the network interference (defined in Section 3) after the completion of phase one and phase two, respectively. The figure shows that, despite the effort in phase one to reduce network interference, the application of phase two to transform the phase one solution to an feasible one raises network interference by about 49–98%, which in a sense nullifies the effort spent in phase one.

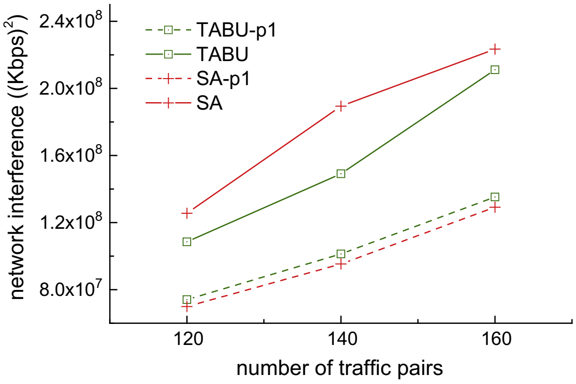

Fig. 2. Increase in network interference after the merge operations are called to resolve the interface constraint violations.

It is a common technique to first relax some constraints on the original problem to find a good approximate solution to the relaxed problem and then modify it, if necessary, to get a feasible solution to the original problem. In fact, the tabu-based algorithm described above exhibits better performance results than some of the existing algorithms [7]. However, it is not clear how much the increase of network interference in phase two is bounded within. In this context we are motivated to consider maintaining or obtaining the solution feasibility in a different way. One of the possible directions could be to follow the two-phase approach as stated above and try to develop another transformation method that causes less increase in network interference in phase two. In this paper, we are interested in other directions that do not, or probabilistically not need to appeal to phase two to get a feasible solution.

In this paper, we solve the channel assignment problem in WMNs using simulated annealing algorithms. Simulated annealing [11] is a probabilistic metaheuristic for finding the global optimum of a given cost function. It belongs to the family of local search optimization techniques with an embedded mechanism to escape from the local optima. When solving a hard problem with a large search space,

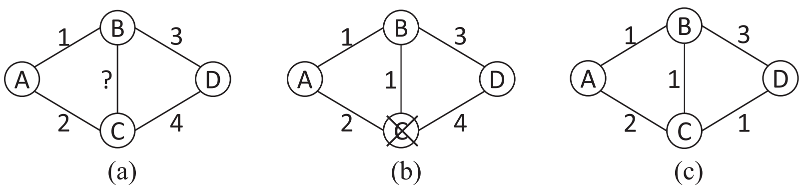

Fig. 1. A sample WMN illustrating the ripple effect.