Observations on Using Simulated Annealing for Dynamic Channel Allocation in 802.11 WLANs

Jiayuan Chen\textsuperscript{1,2} Sverrir Olafsson\textsuperscript{2,3} and Xuanye Gu\textsuperscript{2}

\textsuperscript{1} Department of Electronic and Electrical Engineering, UCL, London WC1E 7JE, UK
\textsuperscript{2} BT, Adastral Park, Martlesham Heath, Ipswich, IP3 5RE, UK
\textsuperscript{3} University of Reykjavík, Ofanleiti 2, 103 Reykjavik, Iceland
E-mail: {jiayuan.chen, sverrir.olafsson, xuanye.gu}@bt.com

\textbf{Abstract}—Reliable channel selection for each Access Point (AP) is essential in setting up and operating densely deployed 802.11 WLANs. The aim with the channel selection is to provide efficient reuse of spectrum and therefore minimize interference and improve user experience. This paper highlights the potential of using Simulated Annealing (SA) to solve the channel selection problem in single-hop multi-AP 802.11b/g networks. The objective is to develop a framework to gauge how well the SA performs in a wireless communication environment. This involves the design of neighboring state transition scheme, annealing schedule selection and the analysis of their impacts to the system performance. The simulation results show that the proposed algorithm is feasible to solve the channel allocation problem and can find the assignments very close to the known optimal solutions under many different network topologies.

I. INTRODUCTION

IEEE802.11 WLANs are predicted to be widely deployed in urban cities to meet the growing demand for wireless data services [1]. 802.11b/g is the most popular choice due to its commercial availability and potential for expansion. However, such capacity expansion is not straightforward because there are only limited numbers of non-overlapping channels (1, 6 and 11) that can be used, in principle, without causing interference between neighboring APs. In many instances of such environment, APs belonging to the same or different vendors can access the common unlicensed frequency band without any restrictions and coordinations between each other. Therefore, it is inevitable that interference cannot be easily controlled. An important task to realize this potential capacity increase is to successfully manage the radio interference problem, that is, to ensure an optimal assignment of radio channels with minimal interference.

In this paper we establish a framework of using Simulated Annealing (SA) to solve the channel allocation problem in the single-hop multiple-AP 802.11b/g networks. Our approach concerns the channel selection by APs, where channels can be selected among orthogonal frequencies. Within this framework, our contribution is to provide a detailed analysis on how to select the parameters of SA in channel allocation to find proper channel assignment which can minimize the total interference. As we demonstrate in the simulation, the total interference is gradually reduced along with the evaluation of proposed algorithm and the resulted channel assignment are very close to the known optimum found by Branch and Bound method [2].

The paper is organized as follows. The following section introduces the general properties of SA. The network model and problem formulation is introduced in Section 3. Section 4 discusses the parameters configuration through theoretical analysis and extensive simulations. Conclusions are given in Section 5.

II. SIMULATED ANNEALING

Simulated Annealing (SA) is a generic probabilistic meta-algorithm for global optimization problems, namely locating a good approximation to the global optimum of a given function in a large search space. It was independently invented by S. Kirkpatrick, C. D. Gelatt and M. P. Vecchi in 1983 [3], and by V. Černý in 1985 [4]. The idea of SA methodology is motivated by the physical process of annealing in metallurgy. In an annealing process, a solid is heated to a high temperature and gradually cooled in order for it to crystalize. At high temperatures the atoms move randomly and at high kinetic energy, but as they are slowly cooled, they tend to align themselves in order to reach the minimum energy state.

SA uses a stochastic approach to direct the search. It allows the system to explore and occupy new states even if the perturbation causes small deteriorations. More precisely, it guides the original local search method in the following way: if $S$ is the current state in the system and $T[S]$ is the corresponding objective function value. Then a perturbation to a new neighbouring state, $S'$, is always accepted if it decreases the objective function, i.e. $\Delta T = T[S'] - T[S] \leq 0$. If on the other hand the new state increases the objective function it will be only accepted with probability which depends on $\Delta T$ and the present temperature,

\[ \Pr[\Delta T, T] = \exp\left(-\frac{\Delta T}{T}\right) \]
where $T$ is an external parameter called the “temperature”. The value of $T$ varies from a relatively large value to a small value close to zero. These values are controlled by a cooling schedule which specifies the initial and instant temperature values at each stage as the iteration progressed. When the temperature is high, stochastic influence is strong, but as the temperature goes down the stochastic factor becomes less important; so the process gradually turns from stochastic behavior to a more deterministic one. It is this feature that helps SA to escape from local optima.

In order to apply the SA into a specific problem, one must specify the state space, the neighboring state transition scheme, the probability transition function and the annealing schedule. These choices can have a significant impact on the method’s effectiveness.

### III. NETWORK TOPOLOGY & PROBLEM FORMULATION

A simplified network model used in this work is shown in Figure 1. The model consists of a number of APs that are randomly deployed in an area. Each AP serves several users that are randomly distributed within a certain communication radius.

![Figure 1 The topology of the network. Mobile users are marked by triangles. Devices are associated with AP closest to them.](image)

In order to accurately model a wireless environment, we have adopted the COST 231 WI (Walfisch-Ikegami) model [5, 6]. It presents a path-loss formula that considers diffraction and reflection of urban building groups. The expression is given by

$$L_{\text{path}} = L_{\text{ref}} + L_{\text{mud}} + 20 \log_{10}(d),$$

where $L_{\text{ref}}$ is the rooftop to street diffraction and scatter loss, which takes city structure into account, such as height of buildings, width of roads, building separation and road orientation. $L_{\text{mud}}$ is the multi-screen diffraction loss and $d$ is the distance between user and AP. A typical parameter setting could be found in [5]. So the received power is

$$P_{\text{rec}} = P_t - L_{\text{path}},$$

where $P_t$ is the transmitted power, which is set to one to all APs in this work.

To apply SA into channel selection problem, an analogy is introduced as:

- The states of the solid corresponding to the feasible channel assignments for all APs.
- The energy of each state corresponding to the total interference in the system.
- The state with minimum energy being the optimal solution.

With this analogy, in the following, the set of all APs is denoted by $A$, and the set of all available channels is denoted by $C$. $|A|$ and $|C|$ are the number of APs and number of available channels respectively. Matrix $\rho: |C| \times |A| \rightarrow \{0,1\}$ is defined in the following way,

$$\rho(i,a) = \begin{cases} 1 & \text{if AP } a \text{ uses channel } i \\ 0 & \text{if AP } a \text{ does not use channel } i \end{cases} \quad (1)$$

Where $i \in \{1,2,\ldots,|C|\}$, $a \in \{1,2,\ldots,|A|\}$. As each AP can only operate in one channel at a time, each column of $\rho$ contains only one ‘1’ and $|C| - 1$ ‘0’. $\Omega_{|A| \times |A|}$ is an interference coefficient matrix. $\omega_{a,b} = 1$ if AP $a$ and AP $b$ use the same channel and they are within each other’s communication range, otherwise $\omega_{a,b} = 0$. For simplicity, we convert $L_{\text{path}}(dB)$ into link gain $G$ ($mW$) that captures the power loss on the path between two APs. Hence the interference experienced by AP $a$, operating on channel $i$, from its neighboring APs $b$ with transmit power $P_t$ is given by

$$I_a(i) = \sum_{b \in \text{NB}(a)} \Omega_{a,b} G_{ab} P_t + \eta_a \quad (2)$$

Where $\text{NB}(a)$ defines a local environment around the AP $a$ but not including $a$ itself, $\eta_a$ is the back-ground noise experienced by AP $a$. The objective of the channel allocation problem is to find a proper channel assignment, $S = (s_1, s_2, \ldots, s_{|A|})$, such that the total interference $\Upsilon[S]$ is minimized, while in vector $S$ each component $s_a$ can take any channel number value. The objective function is formulated as follows:

Minimize $\Upsilon[S] = \sum_{a \in A} I_a(s_a) = \sum_{s_a \in C} \sum_{a \in A \setminus \text{NB}(a)} \Omega_{a,b} G_{ab} P_t + \eta_b$

subject to $\sum_{a=1}^{\lfloor \frac{|A|}{|C|} \rfloor} \sum_{i=1}^{|C|} \rho(i,a) = |A| \quad (3)$
IV. ALGORITHM IMPLEMENTATION

The quality of the results obtained by SA depends on several parameters. In this section, we provide a guideline of how to apply SA into the channel selection problem in 802.11 b/g WLAN. This includes the neighboring state transition design and cooling schedule selection. Simulation results will be compared with known optima to examine the feasibility and scalability properties of the proposed algorithm. For simplicity, in all experiments, the transmit power is set to one unit for all AP. Only 3 channels are available to use. The total interference is normalised by divided by \( |A|^2 \).

A. Neighbouring state transition

The interest in SA to solve combinational optimization problem began with the work of Kirkpatrick et al [3]. It shows that the original SA is quite slow in solving practical problems, therefore in many applications SA has a variety of speedy versions which are strictly problem specific. By applying it in the channel allocation problem, we introduce several neighboring state transition methods. A neighboring state is defined as a new channel assignment produced from its previous assignment by one or few APs selecting new channels. During the transition, the target AP could switch to a channel randomly or to a channel that has not been used by its neighboring APs if possible. In order to complete this transition, we assume the APs can sense the surrounding area and choose any channels available for communication.

We design three transition methods and compare them in terms of total interference. In Algorithm a, we adopted the original version, where only one AP is allowed to switch to a random channel at one time. In Algorithm b, AP is more “intelligent”. It can sense the interference in the channel before swapping, and choose the one that with least interference. While in Algorithm c, we adopted a pure random way, where several APs can simultaneously switch to any other channels in a random way. The details of the algorithms are described as follows:

- In initialization, each AP randomly assigned by a channel.
- Neighbouring state transition:
  a) Algorithm a: In order to find the neighbouring state, one of the APs randomly switches to another channel.
  b) Algorithm b: One of the APs switches to a channel that has the least interference level around the local area.
  c) Algorithm c: Several APs are selected to replace their channels simultaneously.
- Calculate the interference level after transition, if the total interference is less than that of the previous generation, keep the assignment for further evolution, otherwise accept it with a probability of \( \min(1, \exp(\frac{-\Delta T}{T})) \).
- Slightly decrease the temperature \( T \) and repeat step 1 to 3 until the termination condition is met, i.e. the maximum number of attempts is reached.

Figure 2 shows the final channel assignment and total interference level for the individual neighboring state transition scheme under a network with 20 AP. It shows that all the schemes can finally converge to a steady state. However, the final channel assignments are quite different. Algorithm a and c provide better channel assignments than that of Algorithms b, where each AP trying to use different channels from their neighbors. This helps system converge to a lower interference level. The reason why Algorithm b ends up with poor channel assignment is because each AP is always selfish to try to use channel with least interference. This may reduce the diversity of Simulated Annealing algorithm and make the system stuck into a suboptimal solution. Although, it has the fastest convergence rate compared with two other schemes.

B. Cooling Schedule Selection

Cooling schedule controls the way of decreasing temperature \( T \). It has a major impact on convergence rate and solution quality. Generally, if the temperature is decreased quickly then the algorithm converges fast, but final solutions will tend to get worse. On the other hand, slow cooling will make the algorithm slow but give better results. We apply 10 frequently used cooling schedules (shown in Figure 3, left) into two systems, one with 10 AP (centre) and the other with 50 AP (right). Total interference performances are compared in Figure 3.

From Figure 3, we can see that for small systems (10 AP) the total interference almost evolves in the same way as the temperatures cooling down. But for large systems, the slower the cooling the better assignment the system can achieve. This is because when we increase the number of AP in the system, the objective function of channel allocation, which is a discrete function, varies significantly in different simulations. Take a network of 50 APs and 3 available channels as an example, it has 350 possible channel assignments and each assignment is associated with a total interference level. Therefore, slower cooling can provide more opportunities to hit good channel assignment, which also has higher possibilities to find a better solution. However for small system, since the fast and slow cooling can both provide the same result, fast cooling is preferable due to its fast convergence rate.
C. Feasibility

Now, we will examine the solution quality of the proposed algorithm. Based on previous analysis, we will use Algorithm a for neighbouring state transition scheme and cooling 3 as the annealing schedule for the following experiments. First, we present the feasibility of the proposed algorithm, in a small (10 AP) and a large (100 AP) network with APs randomly located in a 1 km² area (shown in Figure 4). We can see that the proposed algorithm can solve the channel allocation problem, even in an extremely dense deployed network (100 AP in 1 km² area). Within the network, all 3 channels have been fully used and spread evenly across the network. Moreover, the algorithm can rapidly converge to a steady state.

D. Compare with optimal solution

To gain some insight into the quality of the solutions delivered by SA we compare it with the optimal solution found by Branch and Bound method. We use a network configuration with 20 APs random deployed in the same area. The optimum solution is given and compared with the first solution we found by SA.
Figure 5 shows the comparison of channel assignment found by two algorithms. SA seems not as good as the optimal solution, since some pairs of neighboring APs are using the same channel. To provide a more systematic evaluation, we compare these two algorithms over 1000 different network configurations. For SA, each AP configuration is run for 1000 times and the mean value is extracted for comparison. The result is shown in Figure 6:
It shows that the channel assignments found by SA are very close to the optimal solutions with network size equal to 20.

E. Scalability

For relatively small problems of the order of 10 APs, it is easy to find a global optimum for the three channel case. The resulting channel allocation leads to the lowest possible total interference. The problem however is that the optimization problem is NP hard and therefore the time it takes to find the optimum solution grows exponentially fast with the size of the problem. Optimal solutions for network sizes up to 30 APs are feasible. However, finding the global optimum for three channels and more than 40 AP within reasonable time does not seem to be possible.

In this section we will investigate the scalability of our proposed algorithm with different network sizes. The simulation will be compared with two other algorithms. The base line is a pure random channel selection scheme. In this scheme, any APs can choose whatever channels they want to transmit on regardless the interference in the system. The upper bound is the global optimal solution, which is found by Branch and Bound method. It can find the same solution that of exhaustive search but with less complexity in large networks. Due the reasons we mentioned above, the maximal network size for comparison will be 30 AP. With each size 1000 different AP configurations will be tested. Except global optimum, each algorithm will repeat 1000 times for every configuration to get a mean value for comparison. Final total interference level is divided by the number of APs to get a measurement as mean minimum interference per node.

From Figure 7, we can see that the proposed algorithm has a very good scalability compared with other algorithms. The total interference achieved in each network size is very close to that of optimal solution. For pure random interference level is increased exponentially against the network size.

V. CONCLUSION

This paper has highlighted the application potential of using Simulated Annealing to solve the dynamic channel selection problem in single-hop multi-AP 802.11 WLANs. Extensive theoretical analysis and experimental simulations have been carried out to provide an insight on how to adjust various parameters to apply this optimization strategy. The simulation results are shown to very close to the optimal solutions.

VI. REFERENCES