# Quasi-static Channel Assignment Algorithms for Wireless Communications Networks

Frank Yeong-Sung Lin
Department of Information Management
National Taiwan University
Taipei, Taiwan, R.O.C.
Tel: +886-2-363-8423 Fax: +886-2-758-4773
Email: yslin@im.ntu.edu.tw

**Abstract**

Quasi-static channel assignment algorithms for wireless communications networks are proposed and examined in this paper. More specifically, for (i) a given number of available channels (ii) locations of the cells and co-channel interference relations among the cells and (iii) channel requirement of each cell (to ensure that the call blocking probability associated with each cell not exceed a given bound), we attempt to find a feasible channel assignment policy on a quasi-static basis to satisfy the channel requirement for each cell without using the same channel for any adjacent cell pair to avoid co-channel interference. This satisfiability algorithm can also be applied with the concept of bisecting search to minimize the number of total channels required. This channel assignment problem in wireless communications networks can be considered as a multi-coloring problem. We specify an integer programming formulation of the problem, which leads to the development of an efficient and effective channel assignment heuristic. In the computational experiments, the proposed algorithm calculates optimal solutions for test networks with up to 100 cells in minutes of CPU time on a PC.

## 1. Introduction

One of the fundamental issues in wireless communications is how to allocate the limited bandwidth available among various cell sites. This is referred to as the channel assignment (or allocation) problem. Although various dynamic channel assignment (DCA) schemes [1] [2] [3] [4] [5] have been proposed (to increase the channel efficiency due to mobility and unbalanced traffic demands), the majority of current systems still employ static assignment (or fixed channel assignment (FCA)) schemes due to simplicity of implementation and ease of operation. In this paper, we consider quasi-static channel assignment (QCA) algorithms. The term quasi-static refers to the fact that the algorithm can be used to re-assign channels on a periodic basis such as daily/hourly to take into account day-to-day/hour-to-hour variation of the traffic demands. In such a system, a set of nominal channels is assigned to each cell (perhaps daily/hourly based on expected traffic demands of the next day/hour), and the same set of channels is reused some “co-channel reuse distance” away. Once the channels are assigned, the operation of the system is the same as the FCA system where an arriving call can only be served by the nominally assigned channels. If all the nominal channels are in use, new calls are blocked.

Most of the channel assignment methods in the literature are proposed on hexagonal network structures [6] [7] [8] [9] [10] [11] [12] [13], but real wireless communications networks may be far from such regular configurations. Another algorithm design objective considered in this paper is therefore to develop a generic methodology for arbitrary network structures (co-channel interference relations among cells). However, the know optimal channel assignment strategies for regularly structured networks can be used to evaluate the solution quality of the algorithm proposed in this paper.

The problem of channel assignment can be considered as a multi-coloring problem in a graph [14]. This problem can also be described as an optimization problem with the objective function to minimize the total number of colors used so that no two adjacent nodes (nodes with a direct link connecting them) can have any same color. In this paper, we consider the latter problem description for the purpose of better evaluating the effectiveness of the proposed algorithm.

In general, the problem of finding an optimal channel assignment policy is NP-complete. As such, to take into account computation time constratints, instead of attempting to solve the problem optimally we would propose an efficient and effective heuristic algorithm to solve the problem, specially targeted for an quasi-static application. This algorithm design criteria, however, may not be met by a number of existing algorithms.

In this paper, we formulate the problem as an integer programming problem where the objective function is the minimization of the total number of channels required subject to co-channel interference constraints among the cells and the channel requirement constraints for individual cells (to have the call blocking probability of each cell no greater than a given performance objective). This formulation can be used for an integer programming package for optimally solving a small-scale problem and facilitates the development of an efficient and effective heuristic algorithm for solving large-scale problems. In the proposed solution procedure, a cost function associated with the assignment of each channel $i$ to a cell $j$ with respect to another cell $k$ is introduced. In each round of the solution procedure, an upper limit of the channels available is specified. The proposed algorithm then iteratively adjusts the cost functions based upon which the channels are assigned. This process is repeated until either a feasible solution is found or a pre-specified number of iterations is exceeded. In the former case, the upper limit of the channels available is reduced, while in the latter case, the upper limit of the channels available is augmented. Then a new round is initiated until no improvement of the total number of channels required is possible.

In computational experiments, the proposed algorithm is tested on a number of networks with the number of cells ranging from 5 to 100. For all the test cases where the optimal solutions are known, the proposed algorithm calculates such optimal solutions in minutes of CPU time on a PC.

The remainder of this paper is organized as follows. Section 2 gives the problem formulation. The solution approach is described in Section 3. Computational results based on the proposed algorithm are reported in Section 4.

## 2. Problem Formulation

Let $L = \{1,2,...,n\}$ be the set of the cells in the wireless communications network. Let $F = \{1,2,...,f\}$ be the set of the available channels of the system. Let $c_j$ be the channel requirement of Cell $j$ and let $A_{jk}$ be the indicator function which is 1 if Cell $j$ and Cell $k$ shall not use the same channel (due to co-channel interference level) and 0 otherwise.

Let $x_{ij}$ be a decision variable which is 1 if Channel $i$ is assigned to Cell $j$ and 0 otherwise, and let $y_i$ also be a decision variable which is 1 if Channel $i$ is assigned to any cell and 0 otherwise. Then the channel assignment problem for wireless communications networks can be formulated as the following integer programming problem.

$$Z_{IP} = \min \sum_{i \in F} y_i \qquad \text{(IP)}$$

subject to:

$$(x_{ij} + x_{ik}) A_{jk} \le 1 \qquad \forall i \in F, j \in L, k \in L \quad (1)$$

$$\sum_{i \in F} x_{ij} \ge c_j \qquad \forall j \in L \quad (2)$$

$$x_{ij} = 0 \text{ or } 1 \qquad \forall i \in F,\ j \in L \quad (3)$$

$$x_{ij} \le y_i \qquad \forall i \in F,\ j \in L \quad (4)$$

$$y_i = 0 \text{ or } 1 \qquad \forall i \in F. \quad (5)$$

The objective function is to minimize the total number of channels assigned to the cells. Constratint (1) requires that Cell $j$ and Cell $k$ not use the same channel when the corresponding intrerference indicator function $A_{jk}$ is 1. Constratint (2) requires that the total number of channels assigned to each channel $i$ be no less than its channel requirement $c_j$. Constraint (4) requires that when channel $i$ is assigned to any Cell $j$ then $y_i$ shall be 1. Constratints (3) and (5) are integer constratints for the channel assignment variables.

The above channel assignment problem is NP-complete, and therefore we do not expect to develop an efficient optimal algorithm for large-scale problems. Instead, an efficient heuristic algorithm is developed and presented in the following section.

## 3. Solution Procedure

In this section, algorithms for solving the channel assignment problem for wireless communications networks are presented. Two approaches based upon relaxation are first considered. These approaches also provide lower bounds on the optimal objective functions value of (IP). Then another primal algorithm is proposed.

Two relaxation approaches are possible to solve (IP). The first is linear programming relaxation where the integer constraints (3) and (5) are relaxed into $0 \le x_{ij} \le 1$ and $0 \le y_i \le 1$, respectively. From an observation of the linear relaxation of (IP), which is referred to as (IP'), the following lemma is established.

**Lemma 1:** The optimal objective function value of (IP') is $\max_{j \in L} c_j$. In addition, for multiple cells, in an optimal solution to (IP'), $x_{ij}$ and $y_i$ are either 0 or 0.5.

Two implications are drawn from Lemma 1. First, from the second part of Lemma 1, an algorithm based upon linear programming relaxation in conjunction with a rounding scheme may not be effective. Second, from the first part of Lemma 1, the lower bound calculated by linear programming relaxation is typically loose. This lower bound can at least be raised to the generalized maximum clique defined as the maximum cligue where the weight of each node, instead of 1, is changed to the corresponding number of channels required.

The second relaxation approach is Lagrangean relaxation. In applying this approach, a number of complicating constraints of (IP) are first identified. They are then multiplied by Lagrangean multipliers and added to the objective function. This process is referred to as dualizing the complicating constraints. We choose to dualize Constraints (1) and (4), and construct the following Lagrangean relaxation problem (LR).

$$Z_D(u,v) = \min \sum_{i \in F} y_i + \sum_{i \in F} \sum_{j \in L} \sum_{k \in L} u_{ijk} [(x_{ij} + x_{ik}) A_{jk} - 1] + \sum_{j \in L} \sum_{i \in F} v_{ij} (x_{ij} - y_i)$$

$$= \min \sum_{i \in F} (1 - \sum_{j \in L} v_{ij}) y_i + \sum_{j \in L} \sum_{i \in F} x_{ij} [\sum_{k \in L} (u_{ijk} + u_{ikj}) A_{jk} + v_{ij}] \quad \text{(LR)}$$

subject to:

$$\sum_{i \in F} x_{ij} \ge c_j \qquad \forall j \in L \quad (6)$$

$$x_{ij} = 0 \text{ or } 1 \qquad \forall i \in F, \forall j \in L \quad (7)$$

$$y_i = 0 \text{ or } 1 \qquad \forall i \in F. \quad (8)$$

(LR) can be decomposed into two independent and easily solvable subproblems where only the $y$ decision variables are involved in the first subproblem and only the $x$ decision variables are involved in the second subproblem.

According to the weak Lagrangean duality theorem, for any $(u,v) \ge 0$, $Z_D(u,v)$ is a lower bound on $Z_{IP}$. The following dual problem (D) is then constructed to calculate the tightest lower bound.

$$Z_D = \max_{u, v \ge 0} Z_D(u,v). \qquad \text{(D)}$$

To solve the dual problem (D), the subgradient method is applied.

It can be shown that in this problem the best lower bound calculated by using Lagrangean relaxation is equal to that calculated by using linear relaxation. As discussed previously, the lower bounds are typically loose.

To obtain primal feasible solutions, it is possible to use solutions to the Lagrangean relaxation problems. More precisely, at each iteration of solving the dual problem, a Lagrangean relaxation problem is solved. If the $x$ decision variables calculated satisfy the constraints in the primal problem, then a primal feasible solution is found. Otherwise, modification on such infeasible primal solutions can be made to obtain primal feasible solutions.

From computational experiments, however, the approach of Lagrangean relaxation is shown to be ineffective in calculating primal feasible solutions. Nevertheless, the procedure to solve the second subproblem in each Lagrangean relaxation problem sheds light on the development of the following primal algorithm, denoted by Algorithm A.

In Algorithm A, a cost function $p(i,j,k)$ associated with the assignment of each channel $i$ to a cell $j$ with respect to another cell $k$ is introduced. In each round of the solution procedure, an upper limit of the channels available is specified. Algorithm A then iteratively adjusts the cost functions based upon which the channels are assigned. This process is repeated until either a feasible solution is found or a pre-specified number of iterations is exceeded. In the former case, the upper limit of the channels available is reduced, while in the latter case, the upper limit of the channels available is augmented. Then a new round is initiated until no improvement of the total number of channels required is possible.

A number of characteristics of Algorithm A are described below. First, when a violation of the co-channel interference constraint for Channel $i$ and cell pair $(j,k)$ is identified, $p(i,j,k)$ where $j > k$ is augmented. This is for the purpose to increase $cost(i,j)$, the cost of assigning channel $i$ to Cell $j$, in the next iteration so that Channel $i$ may be deselected by Cell $j$. If both $p(i,j,k)$ and $p(i,k,j)$ are augmented, then Channel $i$ may also be deselected by Cell $k$, which would result in an undesired oscillation on the channel assignment. This is why only $p(i,j,k)$ where $j > k$ is augmented. However, also from the computational experiments, it is observed that the introduction of randomization of the cost augmentation, specially for large networks, would enhance the efficiency of the algorithm. More specifically, with a given probability, we also augment $p(i,k,j)$ besides $p(i,j,k)$. This has an effect of increasing the stability of the algorithm.

Second, the increment (step size) of $p(i,j,k)$ at each iteration when needed is another important algorithmic parameter. Two criteria are chosen. First, the sequence converges to 0. This is for finer tuning of $p(i,j,k)$'s when the algorithm proceeds. Second, the series diverges. This is to avoid stall of the algorithm. A possible selection of the step sizes to satisfy the above two criteria is $a/(b + c \times$ the iteration counter) where $a$, $b$ and $c$ are constants.

Third, it is frequently observed that in the course of the execution of the proposed algorithm, only a small number of co-channel interference constraints are violated, referred to as the "degree of violation", for a given set of $p(i,j,k)$'s. In such a case, it is often more efficient to directly adjust the channel assignment of those cells involved in the constraint violation by explicit enumeration. In the computational experiments, it is observed that it is efficient to invoke this explicit enumeration procedure for the degree of violation no greater than 8.

The overall algorithm is described as follows.

**Algorithm A:**

Step 1. Initialization.

1(a). Set the threshold of explicit enumeration, $v$, to a pre-specified value.

1(b). Set the maximum number of iterations performed, $M$, to a pre-specified value.

1(c). Set the randomization parameter, $q$, to a pre-specified value.

1(d). Set the iteration counter, *counter*, to 1.

Step 2. $p(i,j,k) = 1 \ \forall\ i \in F, j \in L, k \in L$.

Step 3. $cost(i,j) = \Sigma_{k \in L}\, p(i,j,k) \times A_{jk} \ \forall\ i \in F, j \in L$. $x_{ij} = 0 \ \forall\ i \in F, j \in L$.

Step 4. For each $j$ in $L$, sort $cost(i,j)$'s in an increasing order. Choose the channel(s) associated with the first $c_j$ element(s) of the sorted list and set the corresponding $x_{ij}$ to 1.

Step 5. Check the feasibility of the solution calculated in Step 4. If it is feasible, stop; otherwise, calculate the degree of violation.

Step 6. If the degree of violation is no greater than $v$, adjust the channel assignment of the involved cells using exhaustive search. If a feasible solution is found, stop.

Step 7. Adjust $cost(i,j)$.

7(a). If $(x_{ij} + x_{jk})\, A_{jk} > 1$, then $p(i,j,k) = p(i,j,k) + 1/(counter+1)$ for $j > k$; otherwise, go to Step 8.

7(b). With probability specified by $q$, $p(i,j,k) = p(i,j,k) + 1/(counter+1)$ for $j < k$.

Step 8. If *counter* equals $M$, stop; otherwise, *counter* = *counter* + 1 and go to Step 3.

## 4. Computional Experiments

In this section, computional experiments on Algorithm A are reported. In the experiments, $v$ is set to 8, $q$ is set ot 0.1 and $M$ is set to 10000. Two sets of experiments are performed on a PC to test the efficiency and effectiveness of Algorithm A.

The first set of experiments are perfromed on a number of irregular networks. The co-channel interference relations of the first test network are shown in Figure 1, and the channel requirement of each cell is shown in Table 1. From an observation of the test network, the generalized maximun clique is 19, which is the total number of channels required for Cells $B$, $E$, $F$ and $I$ in Figure 1. This is clearly a lower bound on the totoal number of channels required for the network. When the number of available channels is specified to be 19, Algorithm A calculates a feasible channel assignment in a few seconds. Another experimtnt is performed on the same network where the channel requirement of each cell is 1. When the number of available channels is specified to be 4, which is the minimum number of channels required, Algorithm A calculates a feasible solution in less than one second.

The second irregular network has 12 nodes. The co-channel interference relations are shown in Figure 2. Consider the case where each cell requires $c$ channel(s). Algorithm A calculates a feasible solution in a few seconds when $3 \times c$ channels are available for $c$ ranging from 1 to 10.

The third irregular network is shown in [15]. It has 22 nodes and requires at least 18 channels to satisfy the demand and the co-channel interference constraints. Algorithm A calculates a feasible solution in a few seconds when the number of available channels is specified to be 18.

The next irregular network considered is shown in Figure

3. The channel requirement of each cell is shown in Table 2. It is observed that for such a network with unbalanced load, the following mechanism can enhance the efficiency and effectiveness of Algorithm A. This mechanism is to calculate a generalized maximum clique and fix the channel assignments for the corresponding cells. From an observation of the network, a lower bound on the total number of channels required for the network calculated by the generalized maximum clique is 165. This is the total number of channels required for Cells *F*, *L* and *M* in Figure 4. When the number of available channels is specified to be 165, Algorithm A calculates a feasible channel assignment in minutes.

The second set of experiments are performed on a number of $n\times n$ regular networks (of hexagonal structures) where $n$ ranges from 4 (16 cells) to 10 (100 cells). If all cells require the same number of channels $c$, then it is clear that at least $3\times c$ channels are required to satisfy the demand and the co-channel interference constraints. Algorithm A calculates a feasible solution for each test case in a few seconds when the minimum number of required channels are given.

## References


[1] S. Engel and M. M. Peritsky, “Statistically-optimum Dynamic Server Assignment in Systems with Interfering Servers,’’ IEEE Trans. on Veh. Technol., vol. VT-22, no. 4, Nov. 1973.

[2] T. J. Kahwa and N. D. Georganas, “A Hybrid Channel Assignment Scheme in Large-scale, Cellular-structured Mobile Communication Systems,” IEEE Trans. on Commun., vol. COM-26, no. 4, Apr. 1978.

[3] S. M. Elnoubi, R. Singh, and S. C. Gupta, “A New Frequency Channel Assignment Algorithm in High Capacity Mobile Communication Systems,” IEEE Trans. on Veh. Technol., vol. VT-31, no. 3, Aug. 1982.

[4] M. Zhang and T. P. Yum, “Comparisons of Channel-assignment Strategies in Cellular Mobile Telephone Systems,’’ IEEE Trans. on Veh. Technol., vol. VT-38, no. 4, Nov. 1989.

[5] J. C. Chuang and N. R. Sollenberger, “Performance of Autonomous Dynamic Channel Assignment and Power Control for TDMA/FDMA Wireless Access,’’ IEEE JSAC, vol. 12, no. 8, Oct. 1994.

[6] Papavassiliou, L. Tassiulas, and P. Tandon, “Meeting QOS Requirements in a Cellular Network with Reuse Partitioning,” IEEE JSAC, vol. 12, pp. 1389-1400, 1994.

[7] S. Kim and S. L. Kim, “A Two-Phase Algorithm for Frequency Assignment in Cellular Mobile Systems,” IEEE Trans. on Veh. Technol., vol. 43, pp. 542-548, 1994.

[8] H. Jiang and S. S. Rappaport, “CBWL: A New Channel Assignment and Sharing Method for Cellular Communication Systems,” IEEE Trans. on Veh. Technol., vol. 43, pp. 313-321, 1994.

[9] P. T. H. Chan, M. Palaniswami, and D. Everitt, “Neural Network-Based Dynamic Channel Assignment for Cellular Mobile Communication Systems,” IEEE Trans. on Veh. Technol., vol. 43, pp. 279-287, 1994.

[10] L. J. Cimini, Jr., G. J. Foschini, C.-L. I, and Z. Miljanic, “Call Blocking Performance of Distributed Algorithms for Dynamic Channel Allocation in Microcells,” IEEE Trans. on Commun., vol. 42, pp. 2600-2607, 1994.

[11] M. Zhang and T.-S. P. Yum, “Comparisons of Channel-Assignment Strategies in Cellular Mobile Telephone Systems,” IEEE Trans. on Veh. Technol., vol. 38, pp. 211-215, 1989.

[12] W. Yue, “Analytical Methods to Calculate the Performance of a Cellular Mobile Radio Communication System with Hybrid Channel Assignment,” IEEE Trans. on Veh. Technol., vol. 40, pp. 453-460, 1991.

[13] R. J. McEliece and K. N. Sivarajan, “Performance Limits for Channelized Cellular Telephone Systems,” IEEE Trans. on Information Theory, vol. 40, pp. 21-34, 1994.

[14] W. K. Hale, “Frequency Assignment: Theory and Applications,” Proc. IEEE, vol. 68, pp. 1497-1514, 1980.

[15] J. A. Silvester, “Perfect Scheduling in Multi-hop Broadcast Networks,’’ Proc. ICCC, Sep. 1982.


| Cell | A | B | C | D | E | F | G | H | I | J |
|---|---|---|---|---|---|---|---|---|---|---|
| $c_j$ | 2 | 5 | 4 | 8 | 4 | 6 | 5 | 3 | 4 | 3 |

Table 1 Channel demand for the test network in Figure 1

| Cell | A | B | C | D | E | F | G | H | I | J | K | L | M | N | O | P | Q | R | S | T | U |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $c_j$ | 10 | 13 | 8 | 31 | 15 | 36 | 57 | 28 | 8 | 15 | 18 | 52 | 77 | 28 | 13 | 15 | 8 | 25 | 8 | 8 | 8 |

Table 2 Channel demand for the test network in Figure 3

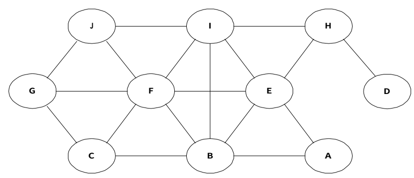


Figure 1 Interference relations of a 10-node network

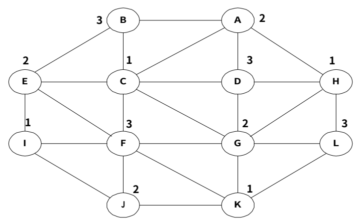


Figure 2 Interference relations of a 12-node network

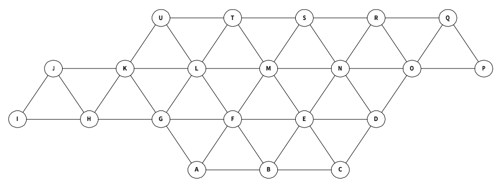


Figure 3 Interference relations of a 21-node network