Journal of ELECTRICAL ENGINEERING, VOL. 54, NO. 7-8, 2003, 169–181

A DECOMPOSITION TECHNIQUE FOR FIXED CHANNEL ASSIGNMENT PROBLEMS IN CELLULAR MOBILE RADIO NETWORKS

Syed Zahid Ali *

The Fixed Channel Assignment (FCA) problem is of major importance in the design of cellular mobile radio networks, and belongs to the class of NP-complete optimisation problems. The known exact solutions to FCA problems are strictly limited by the size of the problem to be solved. In this paper, a new powerful solution approach to the FCA problem is proposed. The underlying principle of the proposed approach is first to transform and then to decompose the original intractable FCA problem into a set of several small-size and weakly interconnected FCA sub-problems by using a special heuristic. This decomposition procedure reduces significantly the complexity of the original FCA problem. The set of small-size FCA sub-problems so obtained is then solved optimally using a sequential Branch and Bound algorithm. The set of solutions obtained for the FCA sub-problems is then the guaranteed optimal solution for the original FCA problem. Computational results obtained using intractable benchmark problems confirm that the proposed approach can be applied successfully to real-world FCA problems.

K e y w o r d s: optimisation, decomposition, graph theory, communication channel

1 INTRODUCTION

In the design of mobile communication networks, the problem of assigning radio channels to Base Stations (BS) is referred to as the Channel Assignment Problem. In the Fixed Channel Assignment (FCA) schemes, each BS is permanently assigned a set of channels for its exclusive use. The optimal solution of the FCA problem is important for the efficient usage of limited radio spectrum and to avoid radio interference conflicts. The optimal assignment of channels, subject to bandwidth and radio interference constraints, is a discrete combinatorial optimisation problem. Since the FCA problem is known to belong to the class of Non-Polynomial (NP) complete problems, the time taken to solve optimally this type of problems scales exponentially with the size of the problem to be solved [1]. Therefore, various heuristic techniques are widely used to obtain an approximate solution in acceptable time. The heuristic techniques used mainly include Tabu Search [2], Neural Networks [3], Genetic Algorithms [4-5], Simulated Annealing [6-7] and Simulated Jumping [8]. The common limitation of heuristic techniques is that they do not provide any theoretical guarantee for global optimality of the given solutions. In order to solve optimally the FCA problem, an analytical technique was proposed earlier in [1]. The FCA problem was developed in terms of a Binary Integer Linear Programming (BILP) problem including the Co-Site Interference (CSI) constraint, Adjacent Channel Interference (ACI) constraint, the Co-Channel Interference (CCI) constraint and the bandwidth constraint. In order to find a true globally optimal solution for the resulting BILP problem, a specially tailored Branch and Bound (B&B) algorithm was used. However, the computational complexity of the exact solution techniques, such as B&B optimisation algorithm, grows exponentially with the size of the FCA problem to be solved. Therefore, the direct application of the B&B algorithm for solving large-scale FCA problems can involve huge computational resources and extended solution times, even with the use of the fastest computers.

In order to enhance the solution capability of the B&B algorithm for solving optimally large-scale FCA problems, an efficient technique is proposed in the present paper. The underlying principle of the proposed technique is to decompose an intractable FCA problem into several small-size, weakly interconnected FCA sub-problems. This decomposition procedure reduces significantly the complexity of the original FCA problem. The set of small size FCA subproblems obtained then can be solved optimally using a specially tailored B&B algorithm [1]. However, the problem of decomposing a FCA problem itself is equivalent to well-known min-cut problem and hence is NP-Complete. Therefore, in order to decompose an intractable channel assignment problem into several small-size FCA sub-problems, a matrix-decomposition based heuristic algorithm is proposed in the paper. The solution of the proposed algorithm is aimed at making it possible to apply a sequential solution procedure to solve optimally and efficiently so-obtained set of small-size FCA sub-problems.

The underlying requirement in all decomposition methods is to develop a scheme capable of partitioning a large system into smaller subsystems such that the elements in the same subsystem are strongly interrelated, but elements included in different subsystems are weakly interrelated. In the context of FCA problems, the decomposition problem involves partitioning a large-scale FCA problem into smaller sub-problems such that the BS's included in

* School of Computing Science,Middlesex University, Bounds Green Road, London N11 2NQ E-Mail: z.ali@mdx.ac.uk

ISSN 1335-3632

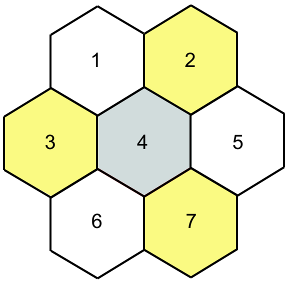

Fig. 1. A 7-cell mobile network. Numbers inside hexagons denote cell indices.

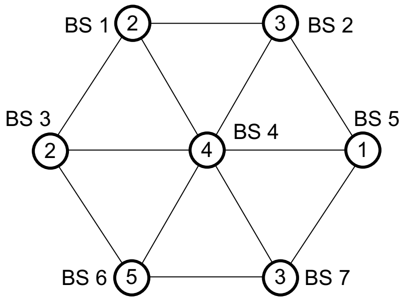

Fig. 2. The Interference Graph for the illustrative network of application example. Circles indicate nodes (cells or BS's), and arcs represent links (or the Co-Channel constraints). Inside each circle is the channel demand. The BS's indices are shown outside the circles.

the same FCA sub-problem are strongly interrelated, but those BS's included in different FCA sub-problems are weakly interrelated. As a decomposed FCA sub-problem is solved, choices of channels for subsequent (next to be solved) sub-problems are reduced. Therefore, the probability that next sub-problem(s) may become infeasible to solve increases. Partitioning a large FCA problem into a set of smaller FCA sub-problems such that this reduction in choices of channels for next sub-problem to be solved is minimised is the objective of the decomposition algorithm put forward in this paper.

Decomposition algorithms have been proposed for large-scale systems in various fields of application. Ford's max flow-min cut algorithm [9] is an efficient technique for partitioning a network into two sub-networks of arbitrary sizes. A heuristic procedure for decomposing a graph into two equal size partitions was proposed in [10]. In [11] an improvement of [10] is proposed. The concept of level gains for partitioning a graph was introduced in [12]. However, the decomposition of an intractable FCA problem into several small-size subproblems involves some special issues that cannot be addressed appropriately using previously known generalized decomposition techniques. This fact contributed to the development of the decomposition technique reported in this paper.

The structure of this paper is as follows. In Section 2, the main parameters of the problem are described. In Section 3, the FCA decomposition problem is presented as a graph-partitioning problem. In Section 4, the decomposition algorithm is presented. Computational results are discussed in Section 5. Finally, some conclusions are presented in Section 6.

2 PROBLEM DESCRIPTION

The electromagnetic compatibility constraints in an N-BS radio network can be expressed in terms of an Interference graph. For an N-BS cellular network, its associated Interference graph [1] $\boldsymbol{G} = (\boldsymbol{V}, \boldsymbol{E})$ is a pair of two finite sets, $\boldsymbol{V} = \{v_1, v_2, v_3, \ldots, v_N\}$ being the set of nodes (or vertices), and $\boldsymbol{E} = \{e_1, e_2, e_3, \ldots, e_N\}$ being the set of arcs (or edges). The arc set $\boldsymbol{E}$ is always a subset of the Cartesian product. $\boldsymbol{V} \times \boldsymbol{V}$ That is, each arc is a pair of nodes $E_k = (v_i, v_j), (i, j) \in \boldsymbol{V}, \forall k \in \boldsymbol{V} \times \boldsymbol{V}$.

In the Interference graph developed, each node represents a BS, the weight of each node represents the bandwidth demand of the BS, and an arc connects two nodes if radio propagation conditions would cause unacceptable interference between the BS's they represent. Note that in subsequent discussion, each arc will be replaced by a pair of oppositely directed arcs. The reason of having a pair of arcs between two nodes is explained later in the paper.

Consider as an application example a 6-BS network shown in Fig. 1. The interference graph for this application example is presented in Fig. 2. In the interference graph shown in Fig. 2, arc between two nodes indicates that the same channel cannot be assigned to the end nodes of the arc. Therefore, the arcs essentially represent the Co-Channel Interference (CCI) constraints.

In order to explain some concepts associated with the problem of decomposing a FCA problem, the application example presented in Fig. 1 is used. Its associated interference graph has been partitioned into two smaller sub-graphs. The partitioned graph is presented in Fig. 3. The boundary that partitions two resulting sub-graphs has been marked by a dotted line. The resulting sub-graphs are labelled as G1 and G2. The set of arcs whose end-nodes are lying in different sub-graphs is called the *cut-set* [13] of the partition. Each sub-graph so obtained is used to develop a FCA sub-problem. For the sub-graphs G1 and G2, the associated FCA sub-problems are denoted as P1 and P2 respectively. The solution to the set of FCA sub-problems obtained provides the solution to the original FCA problem. Note that FCA sub-problems developed using the sub-graphs shown in Fig. 3 cannot be solved independently owing to existence of certain links between G1 and G2. Therefore, the interaction between the sub-graphs must be minimised in order to maximise the probability that each FCA sub-problem can be solved with little or no reference to other FCA sub-problems.

In order to explain the importance of minimising the interaction among FCA sub-problems, a decomposed Interference graph is reproduced in Fig. 4. Consider a case where the problem P1 is solved first. As the FCA sub-problem P1 is solved, 11 radio channels are assigned to

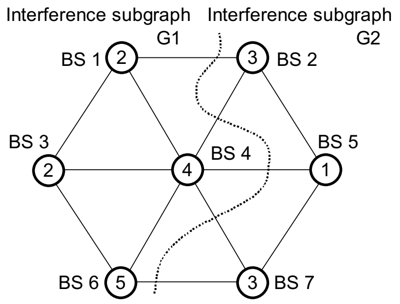

Fig. 3. The partitioning of an Interference Graph into two smaller subgraphs.

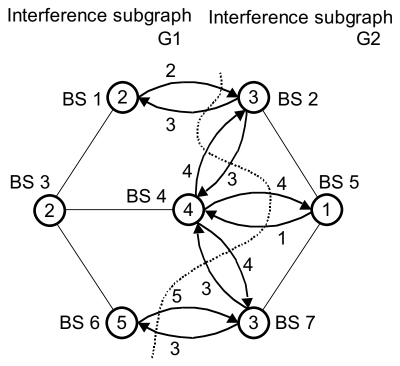

Fig. 4. The partitioning of an Interference Graph into two smaller subgraphs. A directed arc between two nodes denotes the reduction in the number of choices for the next problem to be solved.

boundary nodes (v_1, v_4, v_6) of G1. Owing to the CCI constraints, the full set of these 11 channels cannot be re-used in the FCA sub-problem P2. Therefore, the choices of free channels for subsequent (next to be solved) FCA sub-problem P2 are reduced. The reduction in number of free channels for the next sub-problem to be solved is computed using the set of directed arcs ending in sub-graph G2. From the above discussion, it follows that the probability that the next sub-problems may become infeasible to solve due to reduction in free channel increases with the scale of decomposition. If the FCA sub-problem P2 is solved first, seven channels are assigned to boundary nodes (v_2, v_5, v_7) of G2. This condition is shown by the set of directed arcs ending in sub-graph G1. Therefore, an advantage can be gained if problem P2 is solved first, followed by solving P1. For the situation shown in Fig. 4, solving sub-problem P2 results in a net gain of 4 channels, that is, making 4 additional channels free for assignment in the next FCA subproblem . Since the reduction in choices of free channels for the next to be solved sub-problem is reduced, the sub-problem P2 should be solved first. Therefore, it follows the selection of appropriate sequence for the solution of FCA sub-problems is also important.

Each FCA sub-problem developed is then formulated as a BILP problem [1]. Depending on the strength of interaction among the FCA sub-problems developed, a sequential or parallel B&B algorithm is then used. The mathematical formulation [1] of the FCA sub-problem is discussed in Section 4.

3 THE DECOMPOSITION ALGORITHM

In order to partition a FCA problem, a Matrix Decomposition based Algorithm (MDA) is presented in this section. The algorithm operates on the interference graph developed for a FCA problem, and partitions it into required number of smaller sub-graphs. Generally, the graph partitioning algorithms can be classified into two classes: First, the algorithms that construct a partition form a description of the graph — these are the so-called constructive algorithms; second, algorithm that improve an existing partition — the so-called refinement algorithm. A two-phase technique forms the core of the proposed MDA. In phase-I, a constructive algorithm is used to find a good initial partition. The partition so obtained is improved in phase-II by applying a refinement algorithm involving local transformation of node(s). The improved partition obtained is the optimal solution with respect to initial solution. If it is required to decompose a FCA problem into more than 2 FCA sub-problems, the hierarchical decomposition technique presented in [14] is used. In the following sections, each phase of the MDA is discussed in detail. The MDA algorithm is summarised thereafter.

3.1. The Constructive Algorithm

The aim of constructing an initial solution is to partition a given interference graph into two sub-graphs taking into account only the structural information of the interference graph. The problem of finding an initial solution involves developing two sub-graphs with the minimal number of arcs between them such that each sub-graphs is constrained to a maximum size (in terms of weights of the nodes). In order to build a good initial solution, a matrix-based technique is used. Initially, the Interference graph of a cellular network is described in terms of a matrix. It was established in [1, 14] that a non-directed graph $\boldsymbol{G} = (\boldsymbol{V}, \boldsymbol{E})$ consisting of N nodes and E arcs can be expressed in terms of an N by N symmetric matrix $\mathbf{C}$ by settling the elements c_{ij} associated with the arc $e_{(i,j)} \in \boldsymbol{E}$ that joins v_i to v_j, $i, j \in \boldsymbol{V}$ such that $[c_{ij}] \neq 0$. This matrix is called the compatibility matrix [1, 14-15]. Each non-diagonal element c_{ij}, $i \neq$j in $\mathbf{C}$ represents the minimum separation distance in the frequency domain between two radio channels assigned to the ith BS and jth BS. $c_{ij} = 0$ indicates that BS i and BS j are allowed to use the same channel. On the other hand, $c_{ij} = 1$ signifies that re-use of the same channels at BS i and BS j is prohibited. Therefore, each non-diagonal, unity-valued element c_{ii} in $\mathbf{C}$ represents the Co-Channel constraint. $\mathbf{C}$ is determined a priori by using appropriate radio propagation model. The channel requirements of BS's are defined through a demand vector $\boldsymbol{D}$ [1].

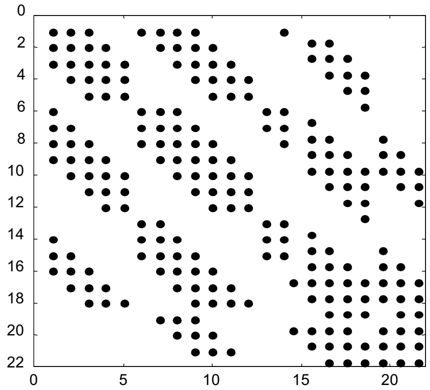

Fig. 5. The bandwidth of the compatibility matrix for problem No. 5 listed in Tab. 1 before application of the RCM algorithm.

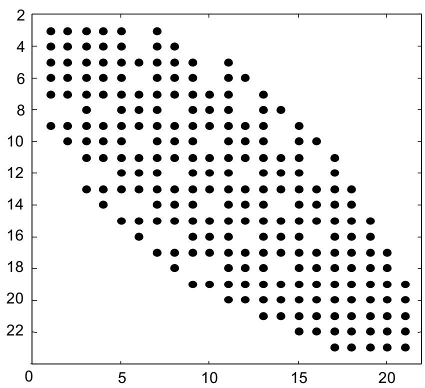

Fig. 6. The bandwidth of the compatibility matrix after applying the RCM algorithm

In order to construct a good initial solution, the $\mathbf{C}$ is initially transformed into a block diagonal matrix in which most of the non-zero elements are clustered around the main diagonal. This is equivalent to developing a matrix with reduced-bandwidth. A reduced bandwidth of connectivity matrix $\mathbf{C}$ implies that matrix $\mathbf{C}$ can be partitioned into a set of two square matrices on the main diagonal such that that there is a minimum number of non-zero elements outside these two square matrices. The two block diagonal matrices of $\mathbf{C}$ are used to define a partition of the interference graph.

As an example, consider the matrix given by (1). In this matrix, two diagonal blocks are identified using perpendicular lines. The upper left block (C1) corresponds to an interference sub-graph of m nodes, whereas the lower right block (C2) defines an Interference subgraph including $N - m$ nodes. The matrix partition shown is optimal with respect to the chosen local optimisation criterion. The local criterion means a criterion that uses only the information in the initial solution.

$$C = \left[\begin{array}{ccc|ccc} c_{11} & \cdots & c_{1m} & c_{1(m+1)} & \cdots & c_{1N} \\ \vdots & \vdots & \vdots & & & \\ c_{m1} & \cdots & c_{mm} & c_{m(m+1)} & & c_{mN} \\ \hline & & & c_{(m+1)(m+1)} & & \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ c_{N1} & \cdots & c_{Nm} & c_{N(m+1)} & \cdots & c_{NN} \end{array}\right]_{N\times N} \tag{1}$$

The problem of clustering nodes closely related to each other in a graph can be regarded as finding a permutation matrix $\mathbf{C}$ resulting in a block diagonal matrix $\mathbf{PCP}^\top$. As discussed earlier, the clusters are defined by the identifiable blocks on the diagonal of the matrix $\mathbf{PCP}^\top$. The simplest methods for re-ordering the nodes of a graph $\boldsymbol{G}$ are the so-called envelope or profile methods. The Reverse Cuthill-McKee (RCM) re-ordering algorithm [16-17] is an efficient profile reduction method. The application of the RCM algorithm to network decomposition problems was initially reported in [18]. Its applicability to FCA decomposition problems has been demonstrated in [19].

Following [17], the bandwidth of C is define as follows:

$$\beta(\mathbf{C}) = \max\{|j - j'|\,,\ \forall j, j' \text{ where } c_{jj'} \neq 0\}\,. \tag{2}$$

The profile or envelope size of $\mathbf{C}$ is defined by the following expression:

$$|Env(\mathbf{C})| = \sum_{j=1}^{N} \beta_j(\mathbf{C})\,. \tag{3}$$

If a linear insertion is used for ordering a list of nodes, the computational time complexity of the RCM is bounded by $0(m|E|)$, where m is the maximum degree of any node and $|E|$ is the number of links (arcs). As an example, in Fig. 5 the original compatibility matrix developed for the problem No. 5 listed in Tab. 1 is shown.

The new compatibility matrix obtained after applying the RCM algorithm to original $\mathbf{C}$ is shown in Fig. 6. Figures 5, 6 illustrate the spread of matrix elements around the main diagonal. It can be seen from Fig. 6 that most elements of $\mathbf{C}$ are clustered around the diagonal. This clustering of elements around the main diagonal increases the probability that the matrix can be decomposed into two sub matrices with minimal links between them. In order to decompose the compatibility matrix, initially a refined version of a technique [19] is used. The technique aims at reducing the undesirable interaction between tentative sub-matrices while satisfying size constraint. Note that non-diagonal square sub-matrices of $\mathbf{C}$ define the cut-set for the decomposed C. These non-diagonal sub-matrices are called cut-set sub-matrices. Therefore, the cut-set sub-matrices describe the interaction among various elements of the $\mathbf{C}$. Following the discussion in [14, 19], the Min-Cut (mc) value using the reordered Compatibility matrix is determined using the following set of expressions:

Minimum-Cut:

$$mc = Min\left(\frac{\sum\limits_{j1=1}^{N1}\sum\limits_{j2=1}^{N2}(A_{j1j2}\cdot B_{j2})}{\sum\limits_{j1=1}^{N1}\sum\limits_{j2=1}^{N2}(A_{j1j2}^{\max}\cdot B_{j2}^{\max})}\right) \tag{4}$$

subject to:

$$\sum c_{ii} \le \omega_{\max}\,, \quad i \in \boldsymbol{V} \tag{5}$$

$$\sum c_{ii} \ge \omega_{\min}\,, \quad i \in \boldsymbol{V} \tag{6}$$

where mc denotes the value of minimum interaction between matrices $\mathbf{C}1$ and $\mathbf{C}2$, $l_1 l_2$ is the cut-set matrix (right top block of $\mathbf{C}$ given by (1) for $\mathbf{C}1$ and $\mathbf{C}2$, and $A^{\max}_{j1j2}$ and $B^{\max}_{j2}$ are the normalisation terms given by the following expression:

$$A_{j1\,j2} = \begin{cases} d_{j1} & \text{if } l_{j1}\, l_{j2} \ne 0\,, \\ 0 & \text{Otherwise.} \end{cases} \tag{7}$$

$$B_{j2} = \sum_{j2'=1}^{N_2} \left(C(j2, j2') \cdot d_{j2'}\right) \quad \forall j2 = 1, \ldots, N_2\,. \tag{8}$$

$\omega_{\max}$ and $\omega_{\min}$ denote the upper bound on the size of the clusters formed, and d_j denotes the j^{th} element of the demand vector $\boldsymbol{D}$ [1]. Note that constraints (5) and (6) ensure that cluster formed are balanced in size. Min-cut obtained using Equations (4)–(8) represents a local optima with respect to optimisation criterion chosen. The reordered compatibility matrix is then decomposed into two square matrices. As an example, a decomposed compatibility matrix is given by (1). This decomposition leads to partitioning the associated interference graph into two smaller interference sub-graphs, which in turn define two FCA sub-problems. Since the RCM algorithm aims at minimising the bandwidth of the compatibility matrix without considering the weight of the nodes involved, the initial solution obtained is susceptible to getting trapped in local optima. In order to improve the partition obtained, a refinement algorithm is used in phase-II. In addition, realistic lower and upper bounds are used to control the size of each sub-graph. These bounds help to maintain symmetry in dimensionalities of the resulting FCA sub-problems.

3.2 The Refinement Algorithm

The proposed refinement technique starts from the initial solution of phase-I, and attempts to improve it according to an objective function. It was explained earlier that the solutions of the FCA decomposition problems are affected by the sequence selected to solve the FCA sub-problems developed. This fact makes it difficult to apply the known refinement algorithms in the context of FCA decomposition problems without major algorithmic changes. In order to deal with this challenge, a problem-specific technique is developed. The technique successively swaps node(s) from one partition to the other. This procedure continues until a stopping criterion is met. The refinement technique has amalgamated some of the ideas presented in [20]. However, the proposed new technique is capable of forming sub-graphs of varying sizes, and shifting one or more nodes at a time. The refinement algorithm is designed in such a way that it can also refine more than two clusters simultaneously; provided the set of these clusters can be expressed in terms of a complete graph.

In order to compute the gains achieved by successive swapping of nodes between sub-graphs, linear equations are developed. These equations are referred to as the Direction-Sensitive Gain (DSG) equations. The DSG equations update gains of *effected* nodes, and take into account the changes in graph structure due to inclusion of a directed arc into a cluster.

In order to develop necessary DSG equations, initially some notation is introduced. Let the initial partition obtained consists of two sub-graphs G_1 and G_2. Note that $e_{(i,j)}, i, j \in \boldsymbol{V}$ represents a directed arc having initial node v_i and final node v_j. If arc $e_{(i,j)}$ is included in the cut-set of the partition such that $i \in G_1, j \in G_2$, and the sub-problem P_1 is solved first, then the reduction in free channels while solving P_2 is equal to the weight of its initial node of this cut-arc.

In general, for most Interference graphs developed for real-world FCA problems, the following condition is true:

$$e_{(i,j)} \ne e_{(j,i)}, i, j \in \boldsymbol{V}\,.$$

The sum of weights of arcs having the same initial node v_i and same final node v_j is denoted as follows:

$$e_{i\leftrightarrow j} = e_{(i,j)} + e_{(j,i)}\,. \tag{9}$$

In a solution including two sub-graphs G_1 and G_2, the cost of the associated cut-set is defined by the following expression:

$$T = \sum_{\substack{i \in G_1 \\ j \in G_2}} e_{(i,j)}\,, \tag{10}$$

where $G_1 \bigcap G_2 = \{\Phi\}$.

In the following section, development of the DSG equation is discussed. In order to improve the initial partition, initially a sequence for solving the FCA sub-problems is chosen arbitrarily. In the Fig. 7, the solution sequence is indicated by an arrow.

Initial Gain Computations for node-by-node shifting procedure:

- Consider a node $v_a \in G_1$ shown in Fig. 7. The out-weight E_a of v_a is defined using arcs with end-nodes in the other sub-graph as follows:

$$E_a = \sum_{i \in G_2} e_{(a,i)} \quad i \ne a\,,\ a \in \boldsymbol{V}\,. \tag{11}$$

The in-weight (I_a) of node v_a is defined by the following expression:

$$I_a = \sum_{i \in G1} e_{(i,a)} \quad i \ne a\,,\ i, a \in \boldsymbol{V}\,. \tag{12}$$

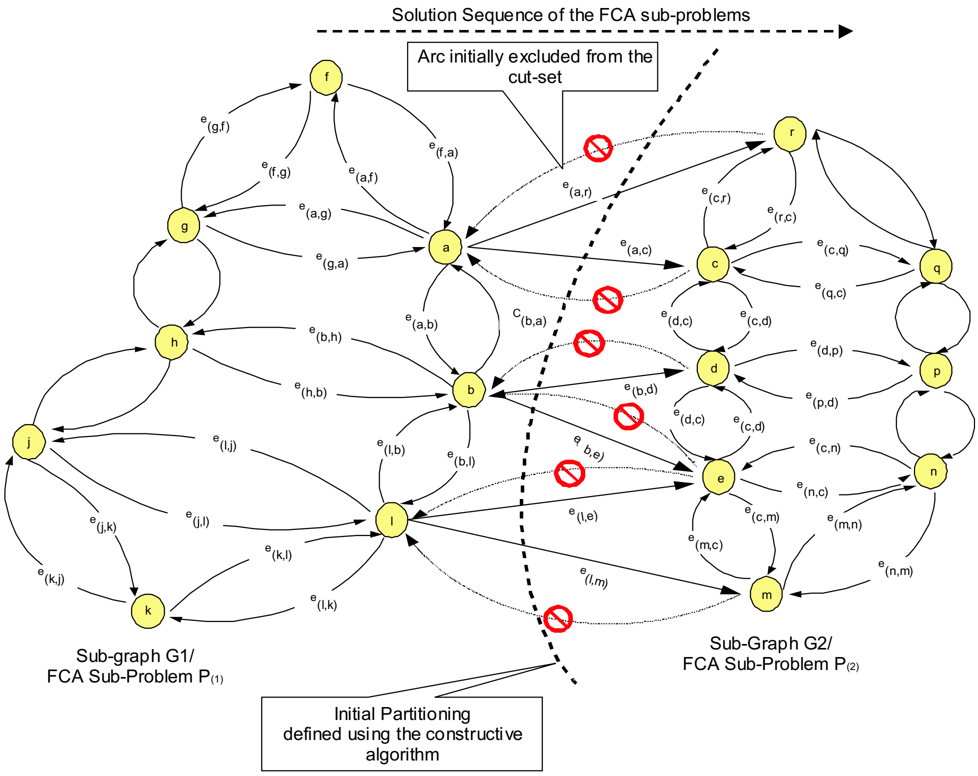

Fig. 7. An initial solution showing a locally optimal partition. The thick dotted line denotes the initial partition. The solution direction is shown by an arrow. The arcs initially blocked for inclusion in the cut-set are also shown.

If the cost of the cut-set without $v_a \in G_1$ is denoted as Z, then the total cost (T) of the cut-set can be expressed as:

$$T = Z + E_a = Z + \sum_{i \in G2} e_{(a,i)} \quad i \neq a\,,\ i, a \in \mathbf{V}\,. \tag{13}$$

If node v_a is shifted to the graph G_2, then the new cost of the cut-set is given by the following expression:

$$\begin{aligned} T' &= Z + I_a \\ &= Z + \sum_{i \in G_2} e_{(a,i)} \quad i \neq a,\ i, a \in \mathbf{V}\,. \end{aligned}$$

Therefore, the net gain obtained by the shifting the node v_a to $G2$ is given as follows:

$$\begin{aligned} Gain = D_a &= T - T' \\ &= \left\{ Z + \sum_{i \in G2} e_{(a,i)} \right\} - \left\{ Z + \sum_{i \in G1} e_{(i,a)} \right\} \\ &= \left\{ \sum_{i \in G2} e_{(a,i)} \right\} - \left\{ \sum_{i \in G1} e_{(i,a)} \right\}. \end{aligned}$$

In general, the gain is computed using the following expression:

$$gain(a) = \sum_{a=(a,j), a \in G1, j \in G2} e_a - \sum_{a=(i,a): a, i \in G1} e_a \tag{14}$$

$$\text{or:} \qquad D_a = E_a - I_a\,. \tag{15}$$

For any subsequent shifting involving a node $v_m \in G_2$, $m \in \mathbf{V}$, the gain is computed using the following expression:

$$D_m = \left\{ \sum_{i \in G1} e_{(i,m)} \right\} - \left\{ \sum_{i \in G2} e_{(m,i)} \right\}.$$

In general:

$$gain(a) = \sum_{a=(i,a), i \in G1, a \in G2} e_a - \sum_{a=(a,i): a, i \in G2} e_a \tag{16}$$

$$\text{or:} \qquad D_a = I_{a'} - E_{a'}\,.$$

Gain Updating for node-by-node shifting procedure:

Shifting a node from a sub-graph involves updating gains of nodes affected by this shifting. The object of shifting nodes is to improve the value of the cut-set between two subgraphs. Since the approach is heuristic, there is a risk that the minimum cut obtained is a local minima. In order to get out of local minima, negative values of net gains are accepted in the main algorithm, as stated in (37). This helps the technique to explore further the feasible solution space for a better partition. If a better solution is found, it is accepted. Otherwise, the last best solution is used to decompose the graph. However, no guarantee for the optimality of the solution is provided in this methodology. Note that there exists a pair of oppositely directed arcs between two incompatible nodes.

Therefore, a successful gain updating procedure must be able to take this fact into account. In order to update the gain of effected nodes, four different cases are considered, and are listed below:

Case I: A node is moved from G_2 to G_1, and gains of affected nodes in G_1 are updated,
Case II: A node is moved from G_2 to G_1 and gains of affected nodes in G_2 are updated,
Case III: A node is moved from G_1 to G_2 and gains of affected nodes in G_1 are updated,
Case IV: A node is moved from G_1 to G_2 and gains of affected nodes in G_2 are updated.

Case I: Consider a node $v_a \in G_2$ is moved to sub-graph G_1. In this case, the gain of an effected node v_x, $x \in G_1, x \neq a$ is updated using the following expressions: Original gain of v_x before transfer:

$$D_x = E_x - I_x\,. \tag{17}$$

After shifting of node v_a, the updated gain of node v_x is given as follows:

$$D'_x = \left\{ \sum_{i\in G_2} e_{(x,i)} - e_{(x,a)} \right\} - \left\{ \sum_{i\in G_1} e_{(i,x)} + e_{(a,x)} \right\},$$

or

$$D'_x = \left\{ \underbrace{\sum_{i\in G2} e_{(x,i)}}_{Ex} - \underbrace{\sum_{i\in G1} e_{(i,x)}}_{Ix} \right\} - \left\{ e_{(x,a)} + e_{(a,x)} \right\},$$

$$D_x = \underbrace{\{Ex - Ix\}}_{D_x} - \underbrace{\left\{ e_{(x,a)} + e_{(a,x)} \right\}}_{sum\ of\ weights\ of\ pair\ of\ arcs} \tag{18}$$

$$D'_x = D_x - e_{(x\leftrightarrow a)}\,. \tag{19}$$

In order to deal with any successive shifting involving a node $v_a \in G_2$, $a' \neq a$ to G_1 affecting the node v_x, $x \in G_1$, $x \neq a'$, the gain is computed as follows:

$$D''_x = D'_x - e_{(x\leftrightarrow a)}\,. \tag{20}$$

Case II: Consider a node $v_a \in G_2$ is shifted to G_1. In this case, the gains of effected nodes v_x, $x \in G_2$, $x \neq a$ are updated as follows:

$$D'_x = \left\{ \sum_{i\in G1} e_{(i,x)} + e_{(a,x)} \right\} - \left\{ \sum_{i\in G2} e_{(x,i)} - e_{(x,a)} \right\},$$

$$D'_x = \underbrace{\left\{ \underbrace{\sum_{i\in G1} e_{(i,x)}}_{Ix} - \underbrace{\sum_{i\in G2} e_{(x,i)}}_{Ex} \right\}}_{Dx} + \underbrace{\left\{ e_{(x,a)} + e_{(a,x)} \right\}}_{sum\ of\ paired\ arcs} \tag{21}$$

or $D'_x = D_x + e_{(x\leftrightarrow a)}$.

In order to deal with any subsequent shifting of a node $v_{a'} \in G_2$, $a' \neq a$ to G_1 affecting the node v_x, $x \in G_2$, $x \neq a'$, the gain is updated as follows:

$$D''_x = e_{(x\leftrightarrow a)} + D'_x \tag{22}$$

Case III: Consider a node $v_a \in G_1$ is moved to G_2. In this case, gains of effected nodes v_x, $x \in G_1$, $x \neq a$ are updated using the following expressions:

$$D_x = \left\{ \sum_{i\in G2} e_{(x,i)} + e_{(x,a)} \right\} - \left\{ \sum_{i\in G1} e_{(i,x)} - e_{(a,x)} \right\},$$

$$D'_x = \underbrace{\left\{ \underbrace{\sum_{i\in G2} e_{(x,i)}}_{Ex} - \underbrace{\sum_{i\in G1} e_{(i,x)}}_{Ix} \right\}}_{Dx} + \underbrace{\left\{ e_{(x,a)} + e_{(a,x)} \right\}}_{sum\ of\ paired\ arcs}, \tag{23}$$

or $D'_x = D_x + e_{(x\leftrightarrow a)}$.

For any subsequent shifting involving a node $v_{a'} \in G_1$, $a' \neq a$ to G_1, affecting the node v_x, $x \in G_1$, $x \neq a'$, the gain is updated using the following expression:

$$D''_x = D'_x + e_{(x\leftrightarrow a')} \tag{24}$$

Case IV: Consider a node $v_a \in G_1$ is moved to G_2. The gain of the effected nodes v_x, $x \in G_2$, $x \neq a$ is updated using the following expression:

$$D_x = \underbrace{\left\{ \underbrace{\sum_{i\in G1} e_{(i,x)}}_{Ix} - \underbrace{\sum_{i\in G2} e_{(x,i)}}_{Ex} \right\}}_{Dx} - \underbrace{\left\{ e_{(x,a)} + e_{(a,x)} \right\}}_{Sum\ of\ paired\ arcs} \tag{25}$$

$D_{x'} = D_x - e_{(x\leftrightarrow a)}$.

In order to deal with any subsequent shifting of a node $v_{a'} \in G_1$, $a' \neq a$ to G_2 affecting the nodes v_x, $x \in G_2$, $x \neq a'$, the gains are updated as under:

$$D''_x = D'_x - e_{(x\leftrightarrow a)}\,. \tag{26}$$

Initial Gain Computations for pairs-of-nodes shifting procedure:

In case where single-node shifting shows little improvement in the value of cut-set, the proposed heuristic switches to shifting pairs of nodes between sub-graphs. Initially, the gains of pairs of border nodes connected through a common arc are computed. In addition, the gains of pairs of nodes not lying on the border, but whose either node is an end-node of a cut-arc, are also computed.

Table 1. Problem specifications and computational results

Problem Number (1)	Total no. of BS's in network (2)	No. of Channels (3)	Solution time for global FCA problem (sec.) (4)	Cut Value obtained decomposition problem (5)	No. of BS in the FCA sub-problem No. P1 (6)	P2 (7)	P3 (8)	Objective value for FCA problem (9)	Solution time (sec) for decomposed FCA problem (10)
1	1225	45	263	210	650	575	NA	0	125
2	324	45	10	111	150	174	NA	0	03
3	21	533	09	769	8	7	6	0	04
4	21	533	09	395	10	11	NA	0	04
5	21	309	310	891	10	5	6	0	167
6	58	29	35	44	25	33	NA	0	22
7	58	29	35	88	20	16	22	0	14
8	21	533	8	259	9	12	NA	0	45

Considering the partition shown in the Fig. 7, the initial gain of a pair of border nodes $v_a, v_b \in G_1$ is computed as follows:

$$D_{(a,b)} = \underbrace{\Big\{\sum_{a\in G_1, x\in G_2} e_{(a,x)} - \Big(\sum_{i\in G_1} e_{(i,a)} - e_{(b,a)}\Big)\Big\}}_{gain(v_a)} + \underbrace{\Big\{\sum_{b\in G_1, x'\in G_2} e_{(b,x')} - \Big(\sum_{i'\in G_1} e_{(i',b)} - e_{(a,b)}\Big)\Big\}}_{gain(v_b)},$$

or
$$D_{(a,b)} = \Big\{\sum_{\substack{a\in G_1\\ x\in G_2}} e_{(a,x)} + \sum_{\substack{b\in G_1\\ x'\in G_2}} e(b,x')\Big\} - \Big\{\sum_{i\in G_1} e_{(i,a)} + \sum_{i\in G_1} e_{(i,b)}\Big\} + \Big\{e_{(a,b)} + e_{(b,a)}\Big\}, \quad (27)$$

$$D_{(a,b)} = (E_a + E_b) - (I_a + I_b) + (e_{a\leftrightarrow b}).$$

In general, the initial gain for a pair of nodes is computed as follows:

$$D_{(a,b)} = \sum_{i=a,b} E_i - \sum_{i=a,b} I_i + e_{a\leftrightarrow b}, \quad v_a, v_b \in G_1.$$

In a similar way, the combined gain of a pair of nodes $v_a, v_b \in G_2$ is computed as follows:

$$D_{(a,b)} = \underbrace{\Big\{\sum_{x\in G_1, a\in G_2} e_{(x,a)} - \Big(\sum_{i\in G_2} e_{(a,i)} - e_{(a,b)}\Big)\Big\}}_{gain(v_a)} + \underbrace{\Big\{\sum_{x'\in G_1, b\in G_2} e_{(x,b)} - \Big(\sum_{i'\in G_2} e_{(b,i')} - e_{(b,a)}\Big)\Big\}}_{gain(v_b)} \quad (28)$$

$$D_{(a,b)} = \sum_{i=a,b} I_i - \sum_{i=a,b} E_i + e_{a\leftrightarrow b}, v_a, v_b \in G_2.$$

The selection of best pair of nodes is made using the following equation:

$$v_{optimal} = \min_{a,b\in B(\boldsymbol{V})} D_{(a,b)} \quad (29)$$

where $B(\boldsymbol{V}) \subseteq \boldsymbol{V}$ is the set of all border nodes of the initial partition. Note that (29) signifies the net gain that can be achieved when both the nodes are moved together to the other sub-graph.

Gain Updating computation for pairs-of-nodes shifting procedure:

Case I: Consider a pair of nodes $v_a, v_b \in \boldsymbol{V}$ is moved from G_1 to G_2. In this case, the gain of affected nodes v_x, $x \neq a \neq b$, $x \in \boldsymbol{V}$ in G_1 is updated using the following expressions:

$$\text{Gain}\{v_x \in G_1\} = \Big\{\sum_{i\in G_2} e_{(x,i)} + e_{(x,a)} + e_{(x,b)}\Big\} - \Big\{\sum_{i\in G_1} e_{(i,x)} - e_{(a,x)} - e_{(b,x)}\Big\} =$$
$$\Big\{\sum_{i\in P_2} e_{(x,i)} - \sum_{i\in P_1} e_{(i,x)}\Big\} + \Big\{e_{(a,x)} + e_{(x,a)}\Big\} + \Big\{e_{(x,b)} + e_{(b,x)}\Big\},$$
$$D'_x = D_x + e_{(x\leftrightarrow a)} + e_{(x\leftrightarrow b)}.$$

For any subsequent swapping of a pair of nodes affecting $v_x \in G_1$, gain is updated as follows:

$$\text{Gain}\{v_x \in G_1\} = D''_x = D'_x + e_{(x\leftrightarrow a')} + e_{(x\leftrightarrow b')}, \quad a', b' \neq a, b. \quad (30)$$

Case II: Consider a pair of nodes $v_a, v_b \in \boldsymbol{V}$ is moved from G_1 to G_2. In this case, the gain of affected nodes

Table 2. Detailed computational results obtained for the FCA subproblems developed.

Problem No	Total no. of PCA subproblems developed	No. of BS's in P1	Solution time (sec.) reported for P1	No. of BS's in P2	Solution time (sec.) reported for P2	No. of BS's in P3	Solution time (sec.) reported for P3	FCA subproblems' solution sequence
(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)
1	2	650	55	575	70	NA	NA	$P1 \to$P2
2	2	150	1.6	174	1.4	NA	NA	P2$\to$P1
3	3	10	1.1	6	1.8	5	1.1	P1$\to$P3$\to$P2
4	2	10	1.8	11	2.2	NA	NA	P1$\to$P2
5	3	10	36	6	42	5	89	P1$\to$P2$\to$P3
6	2	25	7	33	15	NA	NA	P1$\to$P2
7	3	15	4	21	4	22	6	P1$\to$P2$\to$P3
8	2	9	2.1	12	2.4	NA	NA	P1$\to$P2

in G_2 is given as follows:

$$\begin{aligned}\text{Gain}\{v_x \in G_2\} &= \Big\{\sum_{\substack{i\in G_1\\ x\in G_2}} e_{(i,x)} - e_{(a,x)} - e_{(b,x)}\Big\}\\ &\quad - \Big\{\sum_{i'\in G_2} e_{(x,i')} + e_{(x,a)} + e_{(x,b)}\Big\},\\ D'_x &= D_x - e_{(x\leftrightarrow a)} - e_{(x\leftrightarrow b)}\,.\end{aligned}$$

For any subsequent shifting of nodes involving nodes $v_x \in G_2$, the gains of effected nodes is updated as follows:

$$\{\text{Gain of node} v_x \in G_2\} = D''x = D'x - e_{(x\leftrightarrow a')} - e_{(x\leftrightarrow b')}\,,\quad a', b' \neq a, b\,. \tag{31}$$

Similar equations can be developed for the other cases. This completes the description on development of DSG equations. An overview of the Approximate Decomposition Algorithm is presented in the following:

3.3 Summary of the Approximate Decomposition Algorithm

Phase-I: The Constructive algorithm

(1) Express the Interference graph of a network in terms of a Compatibility matrix.
(2) Apply the RCM algorithm to the Compatibility matrix to reduce its bandwidth.
(3) Using the new Compatibility matrix, and a constrained optimisation technique, find an initial partition of the Interference graph.

Phase-II: The Refinement algorithm

Initialisation Set the refinement algorithm to node-by-node shifting mode.

Step 1:
IF (first iteration of node-by-node or pair-of-nodes refinement procedure?)
  IF (first iteration of node-by-node shifting procedure):
    THEN compute initial gains D_i, $\forall i \in \mathbf{V}$, using the appropriate DSG equations
    ELSEIF (first iteration of pair-of-nodes shifting procedure):
    THEN compute initial gains $D_{(a,b)}$, $\forall (a,b) \in \mathbf{V}$, using appropriate DSGC equations.
  ELSE Update gains for effected nodes by appropriate DSG equations.

Step 2:

(a) In case node-by-node shifting procedure is used, select the node with the best gain using the following criterion:

$$v_{optimal} = \min_{i\in\mathbf{V}} D_i \tag{32}$$

(b) In case, the pairs of nodes are under consideration for shifting, select the pair of nodes providing the best gain using the following equation:

$$v_{optimal} = \min_{a,b\in\mathbf{V}} D(a,b)\,. \tag{33}$$

If the size of sub-graph including $v_{optimal}$ exceeds a prefixed tolerance limit (30 %), choose $v_{optimal}$ node(s) from the other cluster.

Step 3:

Move $v_{optimal}$ node(s) to the other sub-graph; lock the $v_{optimal}$ node(s) in destination sub-graph in order to prevent its further shifting.

Step 4:

Go to step 1, and repeat the procedure until set of all nodes is exhausted or the feasibility test fails.

Step 5:

As the current iteration of Refinement algorithm terminates, a list of p gains $p \leq N$ gains, $D_1, D_2, \dots D_p$ has been developed.

Choose a sequence of S gains so as to maximise the partial sum (G) as follows:

$$\sum_{i=1}^{S} D_i = G\,. \tag{34}$$

Step 6:

IF $\left\{\sum_{i=1}^{S} Di > 0\right\}$ and

$$\left\{\Delta_{lower} \leq \sum_{j\in G(q)}^{q=1,2} w_j \leq \Delta_{upper}\right\} \tag{35}$$

THEN perform node(s) shifting as determined by (34) save the partition obtained as current solution. repeat algorithm from step 1 of phase-II. (where Δ_{lower} and Δ_{upper} are lower and upper bounds on the size of sub-graphs, expressed in terms of node weights)

ELSEIF $\left\{\sum_{i=1}^{S} D_i \leq 0\right\}$, (36)

THEN save the current solution as the BEST solution. The min-cut so obtained is denoted as G_{best}.

Step 7:

In order to increase the probability of obtaining a true optimal solution, a negative value of gain is allowed is some runs of the algorithm. Hence the following feasibility condition given by (35) is modified as under:

$$\left\{\sum_{\substack{i=1\\ \xi\leq 0}}^{S} D_i > \xi\right\} \&\& \left\{\Delta_{lower} \leq \sum_{j\in P(q)}^{q=1,2} w_j \leq \Delta_{upper}\right\}. \tag{37}$$

The algorithm is executed for a n iterations ($n = 5$ is used in the present research).

Step 8:

IF $G > G_{best}$

THEN the previously saved BEST solution is replaced with the current solution. Go to step 1 of phase II.

ELSEIF $(G \leq G_{best})$&&(current technique used is node-by-node shifting)

THEN switch to pair-of-nodes shifting methodology; GO TO step 1 of phase II

Note while using pair-of-nodes shifting technique, only even number of nodes are allowed to shift between sub-graphs to ensure the integrity of computations.

ELSEIF $(G \leq G_{best})$&&(Current technique used is pair-of-nodes shifting)

THEN The partition cannot be improved further using the current solution sequence. Select the alternative sequence for solving FCA sub-problems, and repeat the refinement algorithm from the step 1.

Step 9:

Select the best solution, which is an optimal partition with respect to the initial partition, and with respect to all solution sequences.

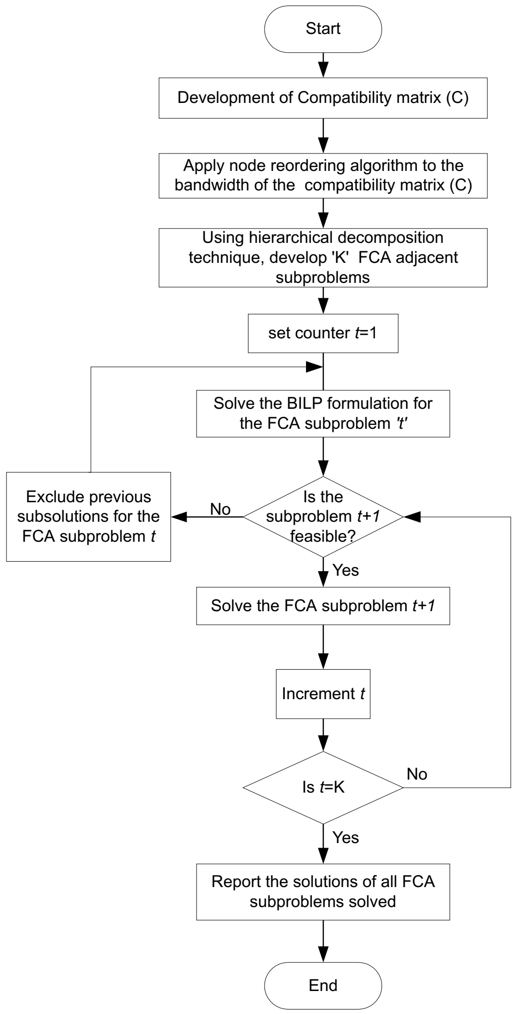

Fig. 8. Flow chart for the proposed decomposition based solution approach.

Each sub-graph so obtained is used to develop a FCA sub-problem. Each sub-problem is then solved using a sequential or parallel Branch and Bound algorithm described later in this paper.

4 MATHEMATICAL FORMULATION FOR THE DECOMPOSED FCA PROBLEM

In this section, the mathematical formulation for the decomposed FCA problem is presented as a BILP problem. Depending on structure of the original Interference graph of the network, the FCA sub-problems developed can be independent, or interconnected. The solution obtained by solving all FCA sub-problems then serves as the solution to the original FCA problem.

A Basic Mathematical Formulation (BMF) for modelling FCA problems was put forward in [1], and is used in the paper.

Consider that the original Interference graph of radio network is partitioned into K interference sub-graphs.

The number of nodes in each sub-graph developed is denoted by N_k, $k = 1, 2, \ldots, K$. Each radio Interference sub-graph gives rise to a FCA sub-problem. The signal interference constraints in a interference sub-graph imply that for any incompatible pair of BS's (j, j'), each channel i and certain adjacent channels i' can be used only once. In context of the k^{th} FCA-sub-problem, this condition can be expressed using the following binary variables:

$$s_i^{j(k)} = \begin{cases} 1 & \text{If } i'^{\text{th}} \text{ channel is assigned} \\ & \text{to } j^{\text{th}} \text{ BS in } k^{\text{th}} \text{ FCA sub-problem,} \\ 0 & \text{Otherwise.} \end{cases} \quad (38)$$

Using the binary variables defined by (38), the CSI, the CCI and the ACI constraints in a FCA sub-problem are defined as follows:

$$s_i^{j(k)} + s_{i'}^{j'(k)} \le 1 \quad \forall (i,j), (i',j') \in \boldsymbol{I} \quad (39)$$

where $\boldsymbol{I}$ is the set of incompatible assignments determined using the Interference graph.

Since each BS may be assigned the required number of channels, therefore the following constraint can be written:

$$\sum_{i \in B} s_i^{j(k)} \le d_{j(k)}, \quad \forall j \in P_k \quad (40)$$

where $d_{j(k)}$ denotes the bandwidth demand of j^{th} BS in the k^{th} FCA sub-problem P_k, and $\boldsymbol{B}$ denotes the set of allocated discrete radio channels, given as follows:

$$\boldsymbol{B} = \{b_1, b_2, b_3, \ldots, b_M\}\,. \quad (41)$$

Note most case the FCA sub-problems developed cannot be solved independently. The interaction among the FCA sub-problems depends on the value of the cut-set obtained for the decomposed FCA problem. Therefore, solving $k-1$ FCA sub-problems first renders a set $B_k \subset \boldsymbol{B}$ of radio channels unusable in certain BS of the next (k) FCA sub-problem to be solved. The cardinality of B_k depends on the cut-set obtained for a partition. Therefore, this variable-prefixing constraint can be written as follows:

$$s_i^{j(k)} = 0\,, \quad i \in B_k\,. \quad (42)$$

Note: cardinality $(B_k) = 0$, if $k = 1$.
The objective function is defined as follows:

$$Z(S) = Min\Big\{\sum_{j=1}^{N_k} d_j - \sum_{j=1}^{N_k}\sum_{i \in B} s_i^{j(k)}\Big\}, \quad k = 1, 2, \ldots, K\,. \quad (43)$$

Since the decision variables are binary and (39) is a linear inequality, the Basic Mathematical Formulation (BMF) represents a Binary Integer Linear Programming (BILP) problem.

This BMF for the k^{th} FCA sub-problem is summarized below:

The objective function:

$$Z(S) = Min\Big\{\sum_{j=1}^{N_k} d_j - \sum_{j=1}^{N_k}\sum_{i \in B} s_i^{j(k)}\Big\}, \quad k = 1, 2, \ldots, K\,. \quad (43)$$

Set of constraints:

$$s_i^{j(k)} + s_{i'}^{j'(k)} \le 1, \quad \forall (i,j), (i',j') \in \boldsymbol{I}\,, \quad (39)$$

$$\sum_{i \in B} s_i^{j(k)} \le d_{j(k)}, \quad \forall j \in P_k\,, \quad (40)$$

$$s_i^{j(k)} = 0\,, \quad i \in B_k\,, \quad (42)$$

$$s_i^{j(k)} \in \{1, 0\}\,. \quad (38)$$

The stronger valid inequalities for BMF are generated using the methodology described in [1]. The Reinforced Mathematical Formulation (RMF) [1] so developed for each FCA sub-problem is then solved using a special B&B 'algorithm [1]. A flow chart showing the overall solution algorithm is presented in Fig. 8.

5 APPLICATION EXAMPLES AND COMPUTATIONAL RESULTS

The proposed decomposition technique has been applied to a range of benchmark FCA problems that have appeared in the literature. The resulting FCA sub-problems are then solved using a B&B algorithm [1]. The mathematical modelling was performed in General Algebraic Modelling System (GAMS) [21], and Cplex [22] was used as an LP solver. Some parts of the proposed decomposition technique were encoded using the C programming language, and Matlab [23].

The problem specifications are summarised in Table 1. Problems No. 6 and Problem No. 7 are taken from [24]. These problems were reported for the area of East Anglia in UK. The associated network consists of 58 BS's, and with uniform bandwidth requirements. Problems No. 3–5 are much celebrated test problems involving significant variations in the bandwidth demand profile, and are taken from [19][25] with some changes. Problem No. 1 and problem No. 2 are large FCA problems, and are used to simulate the conditions reported in [8].

Table 1 includes two types of results. Firstly, it lists the decomposition results obtained for the set of examined benchmark problems. For each problem, the number of BS's in the original FCA problem (Column No. 2), the number of available channels (Column No. 3), the size (in terms of number of BS's) of each FCA sub-problem (Columns No. 6, 7 and 8) are listed. The sub-problems are denoted as P1, P2 and P3 in Table 1. An 'NA' (Not

Applicable) shown in column (8) indicates that the original FCA problem is decomposed into 2 sub-problems. For each decomposed FCA problem, the value of the best cut-set obtained (Column 5) is also listed. Secondly, it lists the objective value (Column 9) and the computational time (Column No. 10) reported by the B&B algorithm for solving the decomposed FCA problems. In order to demonstrate the advantage of using decomposition technique for solving FCA problems, the time reported for solving the original problem globally (that without using the decomposition methodology) is also reported (Column No. 4) . Note that in Table 1, the performance metrics are listed in columns No. 6, 10 and 11. An objective value of zero indicates bandwidth demand of all BS's is met in full without violating any electromagnetic interference constraint.

The detailed computational results for sets of FCA subproblems are presented in the Table II. Problem numbers given in Table 2 correspond to the problems listed earlier in Table 1. For each FCA subproblem developed, the number of BS's included and the computational times are listed. The solution sequence for the FCA subproblems is indicated in column (9), where the notation $Pa \rightarrow Pb \rightarrow Pc$ signifies that the FCA subproblem Pa is solved first, followed by solving Pb, and finally Pc is solved. The decision regarding the solution sequence is made on the basis of the cut-set obtained for a FCA problem. It can be observed from the listed FCA subproblem computational times that the first problem of the solution sequence is always solved in the minimum time. This is due to the availability of the full set of free channels to the first unsolved problem. As the first problem is solved, a number of channels are fixed, and the number of free channels reduce for the 'next-to-be solved' FCA subproblem. Therefore, it takes longer to solve the subsequent FCA subproblems. Since the probability that the 'next-to-be solved' FCA subproblem may become infeasible to solve increases with the scale of decomposition, the development of only 2 or 3 FCA subproblems was considered.

Consider problem No. 6 and problem No. 7 listed in Table 1. As mentioned earlier, these problems are taken from [24]. These problems involve assigning 29 radio channels to 58 BS's. In order to demonstrate the advantage of the decomposition, the problem is first decomposed into two FCA subproblems (listed as Problem No. 6 in Table 1) and then into three FCA subproblems (listed as Problem No. 7 in Table 1). In case of Problem No. 6, a cut-value of 44 was obtained. In case of Problem No. 7, involving three FCA subproblems, a cut-value of 88 is obtained. The increase in cut-value for the problem No. 7 indicates that the probability that 'next-to-be solved' FCA subproblem may become infeasible to solve increases with the scale of decomposition. Each sub-graph is then used to formulate a FCA subproblem. Each FCA sub-problem so developed is then solved using the B&B algorithm [1]. The total solution time reported for the decomposed problem is the summation of the time required to solve each FCA sub-problem. The solution to the set of FCA subproblems then serves as the solution to the original FCA problem. In case of Problem No. 6 and Problem No. 7, the decomposed FCA problems were solved optimally in 22 and 14 seconds respectively. Note that solution time reported for solving Problem No. 6 and Problem No. 7 globally (without using decomposition technique) was 35 seconds. The comparison of the solution times reported shows that the proposed decomposition methodology yields reduced computational times as compared with those reported for solving the problem globally. In addition, these results confirm that the computational time for solving a FCA problem decreases with increase in the scale of decomposition.

Similarly, the problem No. 3 is decomposed into 3 sub-problems with the cut value of 769. The problem No. 4 is decomposed into 2 sub-problems with the cut value of 395. Decomposed FCA Problem No.3 as well as decomposed FCA Problem 4 was in 4 seconds, showing a considerable saving in the computational time. Problem 5 has been decomposed into three sub-problems with cut-set value of 891, and the associated decomposed problem was solved in 167 seconds. Problem No. 8 is taken from [19, 25] with some changes. Using the proposed decomposition methodology, this problem was partitioned into two sub-problems P_1 and P_2 with a cut-set value of 259 . Sub-problems P_1 and P_2 include 9 and 12 BS's respectively. Each sub-problem obtained is formulated as a BILP problem and was solved optimally. Similarly problems No. 1–2 was bisected with cut-set value of 210 and 111. In case of problems No. 1–2, optimal solutions were obtained in shorter time than those reported for solving these problem globally. Problem 4 has been decomposed into two FCA sub-problems using a cut-set value of 395. The solution time reported for its associated FCA sub-problems confirms a clear gain in the computational effort needed to obtain the optimal solution.

In all the problems examined, the application of the decomposition methodologies did not result in infeasibilities. This confirms the validity and utility of the proposed decomposition methodologies for solving the large-scale FCA problems.

For the set of all examined problems, the results obtained demonstrate that the proposed techniques can be used to increase the computational efficiency of the Br&B algorithm for solving the FCA problems.

6 CONCLUSION

Channel Assignment Problem is concerned with the assignment of radio channels to BS's. The optimal assignment of channels, subject to bandwidth and other constraints, is an NP-Complete problem. In order to solve the problem, a deterministic technique was presented earlier in [1]. In order to increase further the efficiency of the proposed deterministic technique, a decomposition framework for large FCA problem is presented in this paper. The underlying principle of the proposed technique is to decompose a large FCA problem into several smaller size

FCA sub-problems. These sub-problems are then solved using a sequential or parallel Branch and Bound algorithm.

In order to decompose a large FCA problem, an efficient heuristic matrix decomposition algorithm (MDA) is put forward. The MDA consists of two phases. The first phase involves constructing a good initial solution. In the second phase, a refinement algorithm is used to improve the initial partition.

By considering a range of benchmark problems, it is shown that the proposed decomposition heuristic can be applied effectively to decompose a given FCA problem into appropriate number of smaller FCA sub-problems. The computational results obtained demonstrate the significantly superior performance of the presented methodologies to other known techniques used for decomposing FCA problems. The computational results obtained from solving the FCA sub-problems confirm that a significant speed-up of the B&B algorithm can be achieved, and that large intractable FCA problems can be solved using the proposed solution methodology.

Acknowledgments

Part of the research reported in this paper was completed under the supervision of Prof. L.F.Turner of Imperial College, London, SW7 2BT, England, UK. Author thankfully acknowledges Prof. L.F.Turner for providing invaluable supervision, guidance and computational resources.

References

[1] ALI, S. Z.—TURNER, L. F.: An Efficient Methodology For Optimal Channel Assignment of Large and Complex Mobile Radio Networks, IEEE 54^{th} Vehicular Technology Conference, vol. 1, pp. 389–393, 7–11 Oct 2001, Atlantic City Convention Centre, Atlantic City, NJ USA.

[2] HAO, J. K.—DORNE, R.—GALINEIR, P.: Tabu Search for Frequency Assignment in Mobile Radio Networks, Journal of Heuristics No. 4 (1998), 47–62.

[3] KUNZ, D.: Channel Assignment for Cellular Radio Using Neural Networks, IEEE Trans. Veh. Tech. **40** No. 1, part 2 (1991), 188–193.

[4] LAI, W. K.—COGHILL, G. G.: Channel Assignment through Evolutionary Optimization, IEEE Trans. Vehicular Tech. **VT-45** No. 1 (1996), 91–96.

[5] BECKMANN, D.—KILLAT, U.: A New Strategy for the Application of Genetic Algorithm to the Channel Assignment Problem, IEEE Trans. Vech. Technology **48** No. 4 (1999), 1261–1269.

[6] DUQUE-ANTON, M.—KUNZ, D.—RUBER, B.: Channel Assignment for Cellular Radio Using Simulated Annealing, IEEE Trans. Vehicular Tech. **VT-42** No. 1 (1993), 14–21.

[7] MATHAR, R.—MATTFELDT, J.: Channel Assignment in Cellular Radio Networks, IEEE Trans. Vehicular Tech. **VT-42** No. 4 (1993), 647–656.

[8] LOCHTIE, G. D.—EIJL, V. C. A.—MEHLER, M. J.: Comparison of Energy Minimizing Algorithms for Channel Assignment in Mobile Radio Networks, Proc. 8^{th} IEEE International Symp. Personal, Indoor and Mobile Radio Communications, PIMRC'97, vol. 3, 1997, pp. 786–790.

[9] FORD, L. R.—FULKERSON, D. R.: Flows in Networks, p. 11, Princeton University Press, Princeton, New Jersey, 1962.

[10] KERNIGHAM, B. W.—LIN, S.: An Efficient Heuristic Procedure for Partitioning Graphs, BSTJ **49** (1970), 291–307.

[11] FIDUCCIA, C. M.—MATTHEYSES, R. M.: A Linear Time Heuristic for Improving Network Partitions, Proc. 19^{th} Design Automation Conference, 1982, pp. 175–81.

[12] KRISHNAMURTY, B.: An Improved Min-Cut Algorithm for Partitioning VLSI Networks, IEEE Transaction on Comput. **C-33** (1984), 438–46.

[13] GARFINKEL, R. S.—NEMHAUSER, G. L.: Integer Programming, Wiley, New York, 1972.

[14] ALI, S. Z.: A Mathematical Programming Approach to Cellular Mobile Radio Network, Ph.D. thesis, University of London, 2002.

[15] GAMST, A.—RAVE, W.: On Frequency Assignment in Mobile Automatic Telephone Systems, Proc. GLOBECOM'82, Miami, FL, 1982, pp. 309–315.

[16] GEORGE, A.: Computer Implementations of the Finite Element Method, Technical Report, STAN-CS-208, Stanford University, 1971.

[17] GEORGE, A.—JOSEPH, W-H L.: Computer Solution of Large Sparse Positive Definite Systems, Prentice-Hall, Inc., Englewood Cliffs, New Jersey 07632,, 1981.

[18] SANGIOVANNI-VINCENTELLI, A.—CHEN, LI-KUAN—CHUA, L. O.: A New Tearing Approach — the node-tearing nodal analysis, Proc. IEEE Intl. Symposium on Circuits and Systems, Phoenix, AR, 1977, pp. 143–48.

[19] GIORTZIS, A.: Channel Assignment in Radio Mobile Network, Ph.D. thesis, University of London, 1997.

[20] KERNIGHAM, B. W.—LIN, S.: An Efficient Heuristic Procedure for Partitioning Graphs, BSTJ **49**, 291–307 yr1970.

[21] BROOKE, A.—KENDRICK, D.: GAMS: A User's Guide, Dec. 1998, GAMS Development Corporation.

[22] CPLEX, IPLOG CPLEX Divison, 889 Alder Avenue, Suite 200, Incline Village, NY 89451 USA.

[23] Matlab, Version 5.3, The Math Works Inc., 3 Apple Hill Drive Natick, MA 01760-2098, USA.

[24] LOCHTIE, G. D.—MEHLER, M. J.: Subspace Approach to Channel Assignment in Mobile Communication Networks, IEE Proceedings on Communications **142** No. 3 (1995), 179–185.

[25] FUNABIKI, N.—TAKEFUJI, Y.: A Neural Network Parallel Algorithm for Channel Assignment Problems in Cellular Networks, IEEE Transactions on Vehicular Technology **VT-41** No. 4 (1992), 430–437.

Received 23 April 2003

Syed Zahid Ali (Dr), received his BE (Electrical) from NED University of Engineering and Technology, Karachi, MES (Telecommunications) degree from University of Sydney, Australia. He received his PhD and DIC degrees from the University of London and Imperial College London in electrical engineering. He has held several technical, research, instructional, operational and managerial posts in Pakistan Telecom, Pakistan and Colt Telecom, Zurich Switzerland, Middlesex University, London and Imperial College, London. He areas of research interest include network design, cellular systems, radio resource allocation, mathematical programming, optimization techniques, graph partitioning techniques, switching systems, operations research and neural networks.