

# Optimal Base Station Clustering for a Mobile Communication Network Design

**Jung Man Hong<sup>1</sup>, Jong Hyup Lee<sup>2</sup> and Soong-Hee Lee<sup>2</sup>**

<sup>1</sup>Business Consultant at Entrue Consulting Partners, LG CNS, Seoul, Korea  
[e-mail: hongjungman@gmail.com]

<sup>2</sup>Department of Information and Communications Engineering, Inje University, Gimhae, Korea  
[e-mail: {icjhlee, icshlee}@inje.ac.kr]

\*Corresponding author: Jong Hyup Lee

*Received February 20, 2011; revised April 12, 2011; accepted April 26, 2011;  
published May 31, 2011*

---

## **Abstract**

This paper considers an optimal base station clustering problem for designing a mobile (wireless) communication network. For a given network with a set of nodes (base stations), the problem is to optimally partition the set of nodes into subsets (each called a cluster) such that the associated inter-cluster traffic is minimized under certain topological constraints and cluster capacity constraints. In the problem analysis, the problem is formulated as an integer programming problem. The integer programming problem is then transformed into a binary integer programming problem, for which the associated linear programming relaxation is solved in a column generation approach assisted by a branch-and-bound procedure. For the column generation, both a heuristic algorithm and a valid inequality approach are exploited. Various numerical examples are solved to evaluate the effectiveness of the LP (Linear Programming) based branch-and-bound algorithm.

---

**Keywords:** Base station, mobile communication network, clustering, branch-and-bound, LP

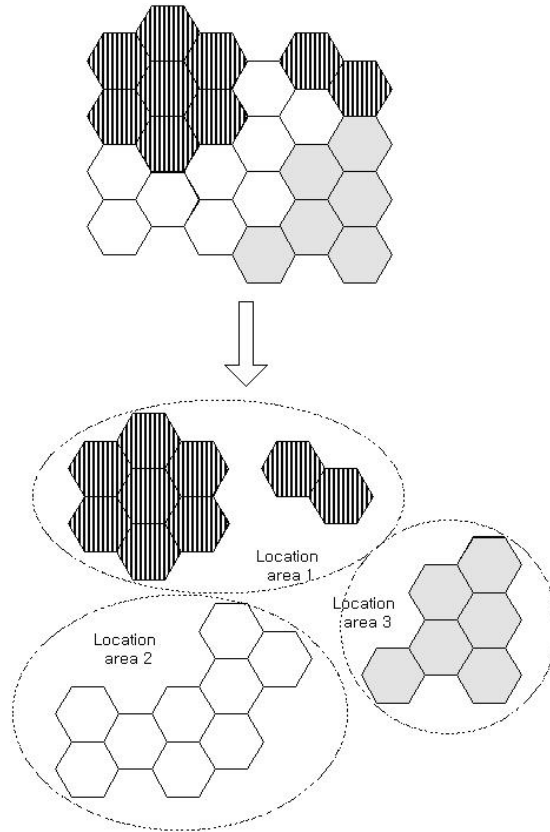
## 1. Introduction

This paper considers an optimal base station clustering problem for designing a mobile communication network consisting of a fixed group of base stations (BSs) covering its service area. The service coverage area of each BS is referred to as a cell that contains a fixed base station and handles a number of mobile subscribers. To take care of each individual communication call, the associated mobile subscriber's location tracking is needed [1].

In cellular systems, each associated network is partitioned into its covering location (or service) areas (each called a cluster) to facilitate the location tracking for all mobile subscribers [2][3][4][5][6][7][8][9][10][11][12]. Thereupon, in this paper we want to derive an optimal clustering algorithm for such a location area design problem in a given mobile communication network where each cluster is restricted to be a complete cluster. A complete cluster is defined as a cluster which does not have any sub-clusters separated (or disconnected) from one another. Thus, this clustering problem is different from any conventional clustering problems in the sense that each cluster should be a complete cluster in its physical topology. These connectivity constraints for the complete cluster are considered to avoid such a situation depicted in Fig. 1 in which the location area 1 has its two sub-clusters of cells being separated from each other.

Whenever a mobile subscriber crosses a border between any two location areas, the network is supposed to be notified to register the change of the subscriber's location so as to incur the location update traffic. This indicates how important the associated location area layout design is for the mobile communication network service management on any frequently occurring mobile location update traffic [1][13]. A paging traffic is also to be generated in the network under the circumstances that as an incoming call attempts to reach any mobile subscriber, all the base stations in the location area where the subscriber is recently registered are initiated to send paging signals. In that network, if the current paging traffic of a location area is too heavy, then the paging service for any incoming call may be delayed [4], which may require for each location area to keep such paging traffic under control below a certain capacity limit.

In the base station clustering problem proposed in this paper, it is assumed that some basic data are given, which includes a set of cells (or base stations) to be partitioned, subscriber movement frequencies between topologically adjacent cells, and paging traffic capacity of each cluster. Therewith, the objective of the proposed problem is to optimally partition the given set of base stations into several clusters such that the sum of the subscriber movement costs (i.e., location updating costs) among clusters is minimized, while the total base station paging traffic of all the clusters is bounded from above and all the base stations in each cluster should be connected with each other as shown in Fig. 1. This base station clustering problem is a kind of the communication network design problem that does not need any real-time decisions. We assume that the subscriber movement frequencies are the input parameters in the network design stage. In real world design problems, the requirement of movement frequencies, which can be converted to cell coverage, is given as parameters when the base station locations are designed [13]. The proposed problem may give rise to a variety of different applications including the location area design problems in mobile telecommunication networks, the facility layout problems, the VLSI design problems, and other geographical area partition problems.



**Fig. 1.** A set of feasible location areas

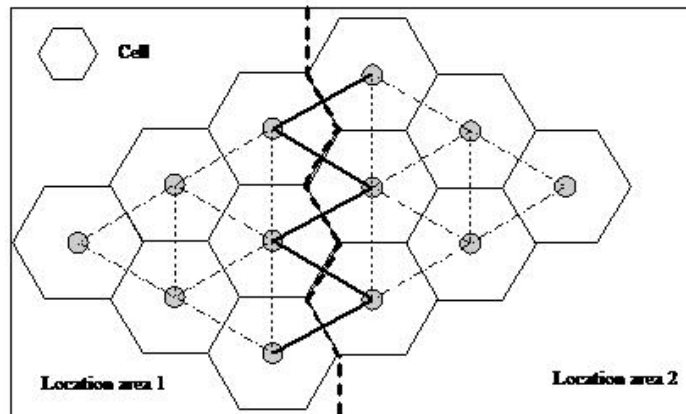
A mobile communication network can be represented as a graph where each base station and any subscriber movement connection between each pair of base stations correspond to a node and an edge, respectively. **Fig. 2** depicts a hexagonal cellular system, illustrating the corresponding graph. In **Fig. 2**, we can see that the network is to be partitioned into two clusters and the cost terms to be included in the objective function are bold lines which represent the subscriber movement costs (i.e., location updating costs).

Correspondingly, the proposed problem can be described as of partitioning the set of nodes  $N$  into the set of disjoint clusters  $L$  with the objective of minimizing the total inter-cluster subscriber movement cost. Specifically, given an undirected graph  $G(N, E)$  with non-negative node weights  $g_i$  representing the paging traffic for each base station for  $i \in N$  and edge costs  $c_e$  for  $e \in E$ , the objective of the problem is to find an optimal partition  $\Gamma = \{W_1, W_2, \dots, W_{|L|}\}$

of  $N$  that minimizes  $\sum_{l \in L} \left\{ \sum_{e \in \delta(W_l)} c_e \right\}$  subject to the constraints such that  $\sum_{i \in W_l} g_i \leq C, \forall l \in L$  and

each partition  $W_l, l = 1, 2, \dots, |L|$  should be a complete cluster, where each partition  $W_l$  can have a set of edges  $(i, j)$ 's connecting it with its adjacent partitions such as  $\delta(W_l) = \{(i, j) | i \in W_l, j \notin W_l, (i, j) \in E\}$ . Recall that in the above graph modeling of the problem,  $N$ , a set of nodes, and  $E$ , a set of links, can represent the set of all cells in a wireless network and the set of all location update events between each pair of partitions, respectively.

It is also noted that if two cells are adjacent, then there exists a link between them. Moreover, each of the non-negative weights  $g_i$  represents the amount of paging traffic incurred by incoming calls to mobile subscribers in the cell  $i$ , so that  $\sum_{i \in W_l} g_i$  represents the total amount of paging traffic incurred in cluster  $l$ . And  $C$  represents the capacity of the paging traffic that each cluster can handle without degrading any communication service quality.



**Fig. 2.** The graph representation of a mobile communication network

The complexity of the proposed problem can be easily shown by reducing the problem to a min-cut clustering problem that is known to be NP-hard [14]. This implies that the proposed problem is also NP-hard.

Several similar works have been reported as in the literature. Johnson *et al.* [14] have studied a general clustering problem by employing a decomposition framework and a column generation scheme. Lim *et al.* [11] have suggested a heuristic approach for a clustering problem for group communication in hexagonal cellular systems. Tcha *et al.* [12] have considered a partitioning problem for a cellular system but without considering any cluster capacity constraints and physical layout constraints. Several other heuristic techniques have been applied to the similar problems, such as tabu search [15], simulated annealing [16], and genetic algorithm [1].

In this paper, we consider the physical topology constraints in mobile cellular networks such that each cluster should be a complete cluster in their physical topology, which has not been explicitly considered yet for the associated network design problems [17][18].

The overall structure of the suggested algorithm is as follows.

The proposed base station clustering problem is formulated as a binary integer programming problem that has a stronger LP relaxation bound than other general integer programming problems. The suggested binary integer programming problem has a huge number of columns. Each column of the suggested binary problem represents any possible clustering of base stations, so it is almost impossible to handle all the columns (clustering) in the problem formulation. Therefore, instead of considering all the columns in the problem formulation, a scheme adding attractive columns is used in the solution procedure. To generate attractive columns, column generation method is suggested for the LP-relaxed binary integer programming problem. With the column generation method, the most attractive column can be generated at each iteration of LP optimization procedure. Both heuristic and branch-and-cut

algorithm are suggested to solve the column generation problem. Heuristic algorithm is applied to get good columns for the column generation problem and branch-and-cut algorithm is applied to have the most attractive column. To guarantee the optimality of the LP relaxed problem, column generation problem must be solved optimally at least one time. Branch-and-cut algorithm is used to find an optimal solution of column generation problem. After the optimal solution of LP relaxed binary problem is obtained, branch-and-bound approach is additionally used if it is any real valued (non-integer) solution.

The remainder of the paper is organized as follows. In Section 2, the proposed problem is formulated as an integer programming problem by use of a decomposition based solution strategy that creates a master problem (a binary integer programming problem) and also a subproblem (column generation problem) to generate columns for the master problem. Then a heuristic procedure and a cutting plane procedure associated with the valid inequality for the subproblem are proposed in Section 3. In Section 4, numerical examples are solved to evaluate the effectiveness of the LP-based algorithm. Finally, a conclusion is drawn in Section 5.

## 2. Problem Formulation

In this section, we derive the integer programming formulation of the proposed problem. Some notations are now introduced to be used throughout the paper.

$N$  : set of nodes,  $i, j \in N$

$|L|$  : maximum number of clusters

$E$  : set of edges between node pairs that are adjacent to each other,  $e \in E$

$c_e$  : location update cost associated with the frequency of subscriber movement along edge  $e$  between cells  $i$  and  $j$  if they do not belong to the same cluster, where  $e = (i, j) \in E$

$g_i$  : paging traffic for node  $i$ ,  $i \in N$

$C$  : paging traffic capacity in each cluster

The proposed problem can be formulated as an integer programming problem that uses a decomposition-based solution strategy to create a master problem, Problem (MP) and also a subproblem, Problem (SP), to generate columns for the master problem [19].

A cluster is to be feasible if the paging and location update traffic at nodes in the cluster satisfy the constraints that the total base station paging traffic of the cluster is bounded from over. Let  $K$  be the set of all the feasible clusters. Then, let  $y_k$  for  $k = 1, \dots, |K|$  be the incidence vectors of all the possible clusters such that  $y_i^k = 1$ ,  $i$ -th element of vector  $y_k$ , if node  $i$  is in the cluster  $k$ , and 0 otherwise, where  $|K|$  can be  $2^{|N|}$  at most. Then the proposed problem can be formulated as Problem (MP) to consider complete clusters.

### Problem (MP)

$$\text{Maximize } \sum_{k \in K} c^k w^k$$

subject to

$$\sum_{k \in K} y_i^k w^k = 1, \quad \forall i \in N \quad (1)$$

$$\sum_{k \in K} w^k \leq |L| \quad (2)$$

$$w^k \in \{0,1\}, \quad \forall k \in K \quad (3)$$

where  $c^k = \sum_{e \in \{(i,j) \mid y_i^k=1, y_j^k=1, (i,j) \in E\}} c_e$  which may denote the profit of the  $k$ -th cluster.

Constraints (1) ensure for each node to belong to exactly one feasible cluster, and constraints (2) mean that the total number of feasible (complete) clusters is not greater than  $|L|$ . The decision variable  $w^k$  represents a feasible cluster  $k$  such that if it is feasible, then it will have the value 1, and 0, otherwise. The linear programming relaxation of Problem (MP) is called Problem (LPMP) which has a strong bound [14][20].

Each column of Problem (LPMP) represents a cell cluster so that there can be at most  $2^{|N|}$  columns, since each cell can be determined as either belonging to a cluster or not. In other words, Problem (LPMP) can have a huge number of columns ( $2^{|N|}$  in the worst case), so that it is impractical to list all the columns. Instead, a repetitive solution procedure starting Problem (LPMP) with a few columns, called a restricted Problem (LPMP), and then adding some “good” columns as required at each repetition stage could be tried to continue until the whole Problem (LPMP) is optimized with all the possible columns [21]. Any small number of such initial columns treated in the restricted Problem (LPMP) may actually provide a good transient solution, though.

Now, for a feasible cluster  $w^k$  with its node set  $V \subset N$ , let each node and the set of edges belonging to the cluster be denoted by  $y_i$  and  $E(V)$ , respectively, which are defined as follows;

$$y_i = \begin{cases} 1, & i \in V \\ 0, & \text{otherwise} \end{cases} \quad \text{and } E(V) = \{(i, j) \mid i, j \in V, (i, j) \in E\}$$

Also, let  $\pi_i$  and  $\lambda$  be the dual variables of the current restricted Problem (LPMP) at a repetition stage, corresponding to constraints (1) and constraints (2), respectively. Then, a subproblem of Problem (MP), called Problem (SP), can be derived by using those dual variables, which shall be optimized at each repetition stage to generate the most attractive columns (i.e., variables making the greatest contribution) to be added to each corresponding restricted Problem (LPMP) [22].

### Problem (SP)

$$\text{Maximize } \sum_{e \in E} c_e z_e - \sum_{i \in N} \pi_i y_i - \lambda$$

subject to

$$z_e \leq y_i, z_e \leq y_j, \forall e = (i, j) \in E \quad (4)$$

$$\sum_{i \in N} g_i y_i \leq C \quad (5)$$

$$\sum_{e \in \delta(A)} z_e > 0, \forall A \subset W, A \neq W, A \neq \phi, W = \{i \mid y_i = 1, i \in N\} \quad (6)$$

$$y_i \in \{0,1\}, \forall i \in N \quad (7)$$

where  $W$  denotes a complete cluster and  $\delta(A) = \{(i, j) \mid i \in A, j \notin A, (i, j) \in E\}$ .

Constraints (4) represent the situation where variable  $z_e$  equals to one only if both the cells  $i$  and  $j$  are included in the same cluster. Constraints (5) describe the paging traffic capacity in the cluster. Finally, constraints (6) represent the connectivity constraints for nodes in the cluster.

At any repetition stage of the solution procedure for Problem (LPMP), if the optimal objective function value of Problem (SP) is positive, then the current optimal solution vector  $\mathbf{y}$  can be selected as a column to be added to the restricted Problem (LPMP) as entering the basis; otherwise, there are no more columns to be added and thus the current solution to the latest restricted Problem (LPMP) is determined as the optimal solution to the Problem (LPMP). If this optimal solution to Problem (LPMP) is integral, then it is to be an optimal solution to Problem (MP). If not, we will additionally implement a branch-and-bound procedure with the current solution in order to obtain the integer solution.

### 3. Solution Approach

#### 3.1 Subproblem Optimization

The subproblem, Problem (SP), is also known to be an NP-hard problem, referring to Johnson *et al.* [14]. In order to guarantee for the solution of the latest restricted Problem (LPMP) to be an optimal solution of Problem (LPMP), the associated final Problem (SP) must be optimally solved by an algorithm, such as the branch-and-cut approach (proposed in this section) other than any heuristic algorithms. This is because if, at any repetition stage, Problem (SP) is solved to result in a negative objective function value, then it will be determined as not giving any attractive column. As tried in this paper, a heuristic procedure may be considered at any intermediate stage for Problem (SP) to find each desired new column. That is, if the associated objective function value of Problem (SP) is positive, then the associated solution vector  $\mathbf{y}$  can be selected as a new column to be added to the restricted Problem (LPMP); otherwise, a cutting plane method and a branch-and-bound method will be considered together to solve Problem (SP) optimally. The cutting plane method will be considered in an LP-based procedure to generate any violated inequality as needed to cut off any fractional solution. Note that the cutting plane is added to cut off any non-integer solution in the LP-relaxed integer problem so that it does not affect the final integer solution but it makes the overall procedure more efficient than using only the branch-and-bound algorithm. If the cutting plane method fails to identify any violated inequality, then a branch-and-bound procedure will be used to determine an optimal solution.

Problem (SP) is now specified equivalently as of finding a subset  $A$  (cluster) of the node set  $N$  that maximizes the term  $\sum_{e \in E(A)} c_e z_e - \sum_{i \in A} \pi_i y_i$  under the constraints that the paging traffic capacity limitation be placed as  $\sum_{i \in A} g_i \leq C$  and the associated graph  $G(A, E(A))$  be connected, where  $E(A)$  is the set of edges which have both end nodes are in the set  $A$ ,  $E(A) = \{(i, j) \mid i, j \in A, (i, j) \in E\}$ .

### 3.2 Heuristic Algorithm

The heuristic procedure for solving Problem (SP) starts with a single node  $i \in N$ ,  $A = \{i\}$ , as the initial current solution, and then in an iterative manner, any node  $j$  that satisfies both the paging traffic capacity constraints and connectivity constraints will be added to the current solution set  $A$ . To satisfy the connectivity constraints, the newly added node  $j$  should be connected with at least one node in  $A$ . The procedure continues until no more nodes can be added.

Let  $F(A)$  be the set of candidate nodes that can be added to the current solution node set  $A$ ,  $F(A) = \left\{ j \mid (i, j) \in E, i \in A, j \in N \setminus A, \sum_{i \in A} g_i + g_j \leq C \right\}$ , and  $Z(A)$  be the objective function value of Problem (SP) associated with the solution set  $A$ ,  $Z(A) = \sum_{e \in E(A)} c_e z_e - \sum_{i \in A} \pi_i y_i$ . Then, the heuristic algorithm called Best Profit or BEP can be derived in a step-by-step procedure.

#### Heuristic algorithm (BEP)

**Step 1.** Let  $I$  be the set of the remaining (unselected yet) nodes, initially,  $I = N$ , where  $A^*$  be the best solution set found so far by the heuristic, and  $A$  be the current heuristic solution set.

**Step 2.** Select an arbitrary node  $i \in I$  and let  $I = I - \{i\}$  and  $A = \{i\}$ .

**Step 3.** Select a node  $j^*$  from the set  $F(A)$  such that  $j^* = \arg \left\{ \max_{j \in F(A)} \left( \sum_{e \in \varphi(j)} c_e - \pi_j \right) \right\}$ , where  $\varphi(j) = \{(i, j) \mid i \in A, (i, j) \in E\}$ .

**Step 4.**  $A = A \cup \{j^*\}$ . If  $Z(A) > Z(A^*)$ , then let  $A^* = A$ .

**Step 5.** If  $F(A) = \emptyset$  and  $I \neq \emptyset$ , then go to Step 2, while if  $F(A) \neq \emptyset$ , then go to Step 3.

**Step 6.** If  $I = \emptyset$ , then stop.

Step 3 in the heuristic algorithm (BEP) is concerned with the node selection rule such that a node is selected as giving the best profit, measured in the (potential) increment of the objective function value when it is added to the node set  $A$  without considering any node weight,  $g_i$ . However, if the node selection rule is modified to consider



$$j^* = \arg \left\{ \max_{j \in F(A)} \left( \frac{\sum_{e \in \varphi(j)} c_e - \pi_j}{g_i} \right) \right\}, \text{ then the algorithm will select a node based on the unit profit}$$

that represents the increment of the objective function value per node weight. Note that the arbitrary selection of node in Step 2 does not affect the final solution. Because an arbitrary node is selected from the set  $I$  for each iteration until the set  $I$  becomes empty, which means all the node will be selected in the BEP algorithm. If a good node is selected at the initial stage, the computation time of the BEP will be shortened but the final solution will be the same.

The better solution between those corresponding to the above two suggested selection rules will be selected as the added node (used as an entering basis) for the associated restricted Problem (LPMP).

### 3.3 Branch-and-Cut Method

To guarantee the optimality of Problem (LPMP), Problem (SP) should be optimized at least one trial of the associated procedure. Therefore, in this section, the cutting plane method and the branch-and-bound method will be used together to solve Problem (SP) optimally.

#### Definition 1

A node set  $S \subseteq N$  that satisfies one of the following two conditions will be called a dependent set : (a)  $\sum_{i \in S} g_i > C$  (Capacity constraint), (b) Nodes in the set  $S$  are disconnected.

However, there exists a connected subset  $D \subseteq N$  of nodes each node of which is connected with any nodes in the set  $S$  (i.e.,  $D \cup S$  being a connected network) but the subset  $D$  does not satisfy the capacity condition,  $\sum_{i \in D \cup S} g_i \leq C$ . (Connectivity constraint)

It is noted that if a node set  $S$  does not satisfy the above conditions, then it will be an independent set. Thereby, we propose a Modified Floyd-Warshall Method (MFWM) procedure to check whether any node set  $S$  is a dependent set. In fact, the check can be made by condition (b) in the Definition 1. In other words, in association with any given disconnected set  $S$ , the MFWM procedure is proposed to find a node set  $D$  with minimal weight, which can make any node pairs in the set  $D \cup S$  to be connected with each other.

For example, given  $S = \{i, j\}$ , the MFWM procedure will find a node set  $D$  such that some nodes in the set  $D \cup S$  can be connected with each of the nodes  $i$  and  $j$  and the value of  $\sum_{i \in D} g_i$  can be at the minimum. Therefore, any two-node pair in a set  $S$  having such a set  $D$  satisfying the condition,  $\sum_{i \in D \cup S} g_i > C$ , cannot belong to the same cluster. Let  $d(i, j)$  denote the minimum value of  $\sum_{i \in D \cup S} g_i$  that represents the required node weight to connect the pair of nodes  $i$  and  $j$  where  $S = \{i, j\}$ . Accordingly, the MFWM algorithm can now be derived as follows.

#### MFWM algorithm

**Step 1.** Initially, set  $d(i, j)$  to  $g_i + g_j$  if  $(i, j) \in E$  or set it to  $\infty$  if  $(i, j) \notin E$ .

**Step 2.** For every node  $s$  and every pair of nodes  $i, j \in N$ , if  $d(i, j) > d(i, s) + d(s, j) - g_i$ , then set  $d(i, j) = d(i, s) + d(s, j) - g_i$ .

**Step 3.** Repeat Step 2 until there is no such update.

Note that if there exist nodes  $i, j \in S$  that satisfy the condition  $d(i, j) > C$ , then the node set  $S$  is a dependent set. Accordingly, Proposition 1 is now derived.

### Proposition 1

*If a pair of nodes  $i, j \in N$  satisfies the condition  $d(i, j) > C$ , then the relation for  $i, j \in N$ ,  $d(i, j) > C$ , is a valid inequality.*

**Proof.** It is straightforward from the definition of the  $d(i, j)$ .

Even though the Proposition 1 can not guarantee the connected solution for Problem (SP), it can reduce the chance to have any solution that violates the constraints (6). Let Problem (RSP) represent Problem (SP) with the valid inequality in Proposition 1 incorporated, and  $RS^*$  be the optimal solution to Problem (RSP). Then,  $RS^*$  will be composed of  $p$  ( $p \geq 1$ ) disjoint and exclusive components (clusters),  $B_1, B_2, \dots, B_p$ , where all the nodes in each component are connected with one another and  $p$  is any positive integer.

### Proposition 2

*If the objective function value of the optimal solution  $RS^*$  is positive, then the objective function value of  $B_t$  for every component  $t$  ( $t = 1, 2, \dots, p$ ) will not be less than zero; that is,  $Z(B_t) \geq 0$ .*

**Proof.** Assume that  $Z(B_t) < 0$  and  $RS = RS^* \setminus B_t$  for a component  $t$  ( $t = 1, 2, \dots, p$ ). Then,  $RS$  satisfies the capacity constraints (5) and also the relation  $Z(RS) > Z(RS^*)$ , which contradicts the hypothesis that  $RS^*$  is optimal.

### Proposition 3

*The optimal objective function value of Problem (RSP) is positive if and only if there exists an optimal solution  $S^*$  to Problem (SP) which results in a positive objective function value  $Z(S^*) > 0$ .*

**Proof.** It is straightforward. Thus, the detail of the proof is omitted.

Based on the results of Propositions 2 and 3, Problem (RSP) can be easily solved, instead of Problem (SP), to generate the entering basis (i.e., attractive added columns) for Problem

(MP). In other words, if the solution of Problem (RSP) produces a positive objective function value and the solution is composed of one component, then the solution will be added as an entering basis of Problem (MP). If the solution of Problem (RSP) gives a positive objective function value but the solution is composed of more than one component, say  $p$  components, then it will form  $p$  columns ( $p$  generated columns) to be added to Problem (MP). In order to solve Problem (RSP), the proposed heuristic algorithm (BEP) will be tried first. However, if the algorithm fails to get any solution that gives a positive objective function value, then an LP-based procedure (i.e., a linear programming relaxation) will be used to solve Problem (RSP). In the LP-based approach, if the optimal solution of the linear programming relaxation of Problem (RSP) is fractional, then every violated inequality in Proposition 1 will be added to Problem (RSP) and the resulting LP relaxation will be solved to get a new optimal LP solution. Note that Problem (RSP) without that constraints in Proposition 1 incorporated is a knapsack quadratic problem (KQP), for which several valid inequalities can be developed [14]. This whole procedure is repeated until either the integral solution is found or there exist no more valid inequalities. In the latter case where no more valid inequalities are found, it is desired to employ a branch-and-bound procedure.

#### 4. Computational Results

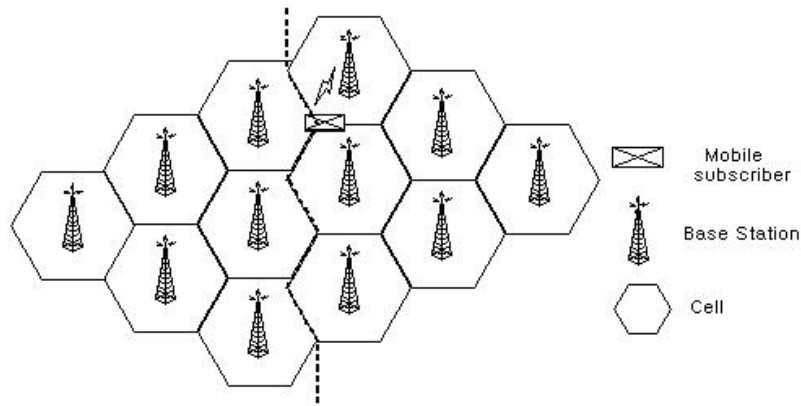
The initial restricted Problem (LPMP) will have columns corresponding to single edge clusters (each single edge cluster being formed only with two adjacent nodes), implying that the total number of columns is equal to the total number of edges. More columns will be generated as the iterative operation of the proposed solution procedure for Problem (SP) continues until Problem (LPMP) is optimized. Note that the heuristic algorithm is used to solve Problem (LPMP). If the heuristic algorithm fails to get any positive objective function value, then the branch-and-cut algorithm will be activated to get the optimal solution of Problem (SP). If the optimal objective function value of the problem is non-positive at the optimal solution, then the solution of the current restricted Problem (LPMP) will be optimal. Moreover, whenever at any intermediate stage of the iterative operation, the optimal solution to Problem (LPMP) is fractional, a simple branch-and-bound procedure will be used to derive an integer solution without generating any more columns. This can of course yield a sub-optimal solution, while it is shown in the numerical tests that only few ones among the tested problems have fractional optimal solutions, and also that the integer solution obtained by the branch-and-bound approach is very close to the solution of the associated linear programming relaxation in terms of the objective value.

To test the effectiveness and efficiency of the proposed whole solution algorithm, several numerical examples for the location area design problem in a cellular mobile network are solved with the input data generated as in Fig. 3 and Fig. 4.

##### Location Update Traffic

Let  $\Lambda(x, y)$  be the population density of mobile subscribers at a point  $(x, y)$  in the two-dimensional vector space of coordinates  $x$  and  $y$ , and  $v_n(x, y)$  be the average velocity of a normal subscriber movement in the direction of the cell-to-cell border, which is represented as a function of the coordinates,  $x$  and  $y$ . Let  $B$  be the cell-to-cell border expressed as a perimeter. Then, the cell crossing rate  $cr$  can be derived as their product measure integrated along the

border,  $\int_B \Lambda(x, y) v_n(x, y) dB$ . Assume that the direction of the subscriber movement is uniformly distributed from  $-\frac{\pi}{2}$  to  $\frac{\pi}{2}$  at a velocity  $v$  [Km/h]. Then,  $v_n(x, y)$  can be simplified as  $v_n(x, y) = \int_{-\frac{\pi}{2}}^{\frac{\pi}{2}} v \cdot \cos(\beta) d\beta = \frac{v}{\pi}$ .



**Fig. 3.** Location update traffic to be incurred by a mobile subscriber crossing the cluster border

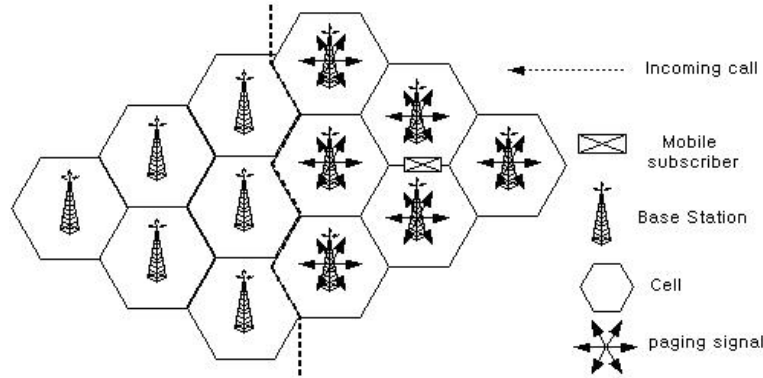
Assume that  $\Lambda(x, y)$  and  $v$  are constant along the associated border and that a hexagonal cell structure is considered. Let  $s$  be the cell side length [Km]. Then, the associated cell crossing rate can be derived as in Plassmann *et al.* [23];  $cr = \frac{\Lambda \cdot v \cdot B}{\pi} = \frac{\Lambda \cdot v \cdot 6s}{\pi}$ . Moreover, the location update rate can be derived from the cell crossing rate as follows: Let  $p_o$  be the probability that a mobile device (i.e., a mobile subscriber) is powered on,  $tr$  be the average traffic intensity per subscriber, and  $m_u$  be the length of the location update message. Then, the location update traffic in cell  $i$  can be derived [23];

$$u_i = cr_i m_u (p_o - tr) = \frac{6\Lambda_i \cdot v_i \cdot s_i}{\pi} m_u (p_o - tr).$$

### Paging Traffic

Let  $\tau$  be the average call holding time. Then, the rate of calls received by a subscriber can be measured as its traffic intensity divided by the average call holding time. Moreover, the number of subscribers in a cell can be measured as the density of people multiplied by the area of the cell. Finally, the area of a hexagonal cell can be measured as  $\frac{3\sqrt{3} \cdot s^2}{2}$ . Let  $p_m$  be the probability that a mobile is not busy, and let  $m_p$  be the length of the paging message [bytes]. Then, the paging traffic in cell  $i$  can be derived as in Plassmann *et al.* [23];

$$g_i = \frac{tr}{\tau_i} \Lambda_i \frac{3\sqrt{3} \cdot s_i^2}{2} m_p p_m.$$



**Fig. 4.** Paging traffic incurred by an incoming call to communicate with the mobile subscriber

The effectiveness and efficiency of the proposed solution algorithm are tested with the networks composed of 25, 49 and 100 cells. The density of subscribers  $\Lambda_i$ , the average traffic intensity per subscriber  $t_i$ , the average velocity  $v_i$ , cell size  $s_i$ , and the average call holding time  $\tau_i$  are all assumed to depend on the cluster. The probabilities of having a subscriber being powered on and of receiving a call by a mobile device are assumed to be 0.5 and 0.3, respectively. The lengths of the location update message and of the paging message are also assumed to be 18 bytes and 8 bytes, respectively [12]. For each of these test problems, twenty instances are generated and solved. To solve these numerical examples, the CPLEX callable mixed integer library is used as an LP solver. An LP solution routine and some other routines in the library are also used for adding constraints and variables. All problems are solved on a PC.

The computational results are shown in **Table 1, 2 and 3**. **Table 1** shows the computational results for test networks with various numbers of cells. The average gap between the solution bound (obtained by LP relaxation) and the solution (found by the proposed algorithm) is very small, less than 0.27% in the 100-cells problem. The CPU time required to solve each of the 49-cells problems does not exceed 3 minutes and is 16.27 seconds on average. The worst case in the CPU processing time takes about 30 minutes for the 100-cells problem instances, while the average CPU time required to solve the problem is about 2.8 minutes. The CPU times shown in **Table 1** are reasonable to solve the network optimization problem.

**Table 1.** Computational results with various size networks

Number of nodes	a	b	c	d
25 Cells	44.53	44.34	0.43	1.60
49 Cells	94.25	93.95	0.31	16.27
100 Cells	195.90	195.38	0.27	168.35

- a : optimal value of Problem (LPMP), LP relaxed problem of MP
- b : optimal value of Problem (MP) from the suggested algorithm
- c : relative ratio of MP to LPMP in percentage
- d : CPU time in seconds for solving the problem

**Table 2** and **3** show the computational results for 49-cells network problem under various capacities of the location area and velocities of subscribers' movements, respectively. Similar to the results in **Table 1**, the average gaps and the average CPU times for the various network problems shown in **Table 2** and **3** are good enough to solve those kinds of network problems.

**Table 2.** Computational results with various capacities of the location area

Capacity of location area	a	b	c	d
10	90.08	89.78	0.33	15.42
15	89.15	88.88	0.30	16.13
20	94.25	93.95	0.31	16.27
25	96.86	96.38	0.50	19.73
30	100.73	100.43	0.30	25.77

**Table 3.** Computational results with various velocities of subscribers' movements

Average velocity of a subscriber movement	a	b	c	d
0.1	81.75	81.60	0.18	13.50
0.5	87.21	86.92	0.33	14.73
1	94.25	93.95	0.31	16.27
5	151.61	151.02	0.39	16.77
10	224.63	223.88	0.33	22.20

## 5. Conclusion

This paper investigates a clustering problem for wireless network design under the constraints that each cluster of the cells (base stations) should not have any sub-clusters separated from each other. It is shown through numerical tests that the LP relaxation of the binary integer programming problem (master problem) formulated for the proposed problem, having exponentially many variables, can be solved efficiently by use of a column generation procedure, and also that an efficient heuristic algorithm is proposed for column generation. Moreover, a valid inequality is found and used to tighten the solution domain of the associated column generation problem. Experimental test results show that the practically large-sized problems can be solved in a reasonable time.

For further study, it may be interested in some extended problems including an adaptive design problem of location area (or cluster) to respond to dynamic traffic change, and a multi-layer location area design problem to accommodate the various classes of subscriber movement velocities.

## References

- [1] Mark V. and Evelyn C. B., "A grouping genetic algorithm for registration area planning," *Omega* vol. 34, no. 3, pp. 220-230, 2006. [Article \(CrossRef Link\)](#)
- [2] Javid T., Albert Y.Z. and Mohsin I., "Fuzzy online location management in mobile computing environments," accepted for publication in *Journal of Parallel and Distributed Computing*, 2011. [Article \(CrossRef Link\)](#)
- [3] Selvan C., Shanmugalakshmi R. and Nirmala V., "Location Management Technique to Reduce Complexity in Cellular Networks," *International Journal of Computer Science Issues*, vol. 7, no. 4, no. 1, pp. 37-40, 2010. [Article \(CrossRef Link\)](#)

- [4] Chew Y.H., Yeo B.S. and Kuan D.C.M., "Sensitivity study of location management area partitioning in cellular communication systems," *Computer Networks*, vol. 51, no. 6, pp. 1555-1573, 2007. [Article \(CrossRef Link\)](#)
- [5] Rajalakshmi K., Prakash K. and Hima M. B., "Hybridizing Iterative local search algorithm for assigning cells to switch in cellular mobile network," *International Journal of Soft Computing* 5(1), pp. 7-12, 2010. [Article \(CrossRef Link\)](#)
- [6] Madhubanti M. Amitava M. and Debashis S., "Integrated Interswitch Cable and Handoff Cost Minimizing Heuristics for Cell-to-Switch Assignment in Wireless Cellular Networks," *IEEE Transactions on Vehicular Technology*, vol. 58, no. 9, pp. 5074-5082, 2009. [Article \(CrossRef Link\)](#)
- [7] Plehn J., "The design of location areas in a GSM-network," in *Proc. of the IEEE 45th Vehicular Technology Conference*, Chicago, USA, pp. 871-875, 1995. [Article \(CrossRef Link\)](#)
- [8] Merchant A. and Sengupta B., "Assignment of Cells to Switches in PCS Networks," *IEEE/ACM Transactions on Networking*, vol. 3, no. 5, pp. 521-526, 1995. [Article \(CrossRef Link\)](#)
- [9] Ian F. Akyildiz and Josep S.M. Ho, "On location management for personal communications networks," *IEEE Communications magazine*, vol. 34, no. 9, pp. 138-145, 1996. [Article \(CrossRef Link\)](#)
- [10] Abutaleb A. and Victor O.K Li, "Location update optimization in personal communication systems," *Wireless Networks*, vol. 3, no. 3, pp. 205-216, 1997. [Article \(CrossRef Link\)](#)
- [11] K. Lim and Y.H. Lee, "Optimal partitioning of heterogeneous traffic sources in mobile communication networks," *IEEE Transactions on Computers*, vol. 46, no. 3, pp. 312-325, 1997. [Article \(CrossRef Link\)](#)
- [12] D.W. Tcha, T.J. Choi and Y.S. Myung, "Location-area partition in a cellular radio network," *Journal of the Operational Research Society* 48, pp. 1076-1081, 1997. [Article \(CrossRef Link\)](#)
- [13] Erdal Cayirci and Ian F. Akyildiz, "Optimal Location Area Design to Minimize Registration Signaling Trace in Wireless Systems," *IEEE Transactions on Mobile Computing*, vol. 2, no. 1, pp. 76-85, 2003. [Article \(CrossRef Link\)](#)
- [14] E.L. Johnson and A. Mehrotra and G.L. Nemhauser, "Min-cut clustering," *Mathematical programming*, vol. 62, no. 1-3, pp. 133-151, 1993. [Article \(CrossRef Link\)](#)
- [15] Tabitha James, Mark Vroblefski and Quinton Nottingham, "A hybrid grouping genetic algorithm for the registration area planning problem" *Computer Communications*, vol. 30, no. 10, pp. 2180-2190, 2007. [Article \(CrossRef Link\)](#)
- [16] Chun-hyun Paik and Samit Soni, "A simulated annealing based solution approach for the two-layered location registration and paging areas partitioning problem in cellular mobile networks," *European Journal of Operational Research*, vol. 178, no. 2, pp. 579-594, 2007. [Article \(CrossRef Link\)](#)
- [17] S. Chopra and M.R. Rao, "The partition problem," *Mathematical Programming*, vol. 59, no. 1-3, pp. 87-115, 1993. [Article \(CrossRef Link\)](#)
- [18] C.E. Ferreira, A. Martin, C.C. Desouza, R. Weismantel and L.A. WOLSEY, "Formulation and valid inequalities for the node capacitated graph partitioning problem," *Mathematical Programming*, vol. 74, no. 3, pp. 247-266, 1996. [Article \(CrossRef Link\)](#)
- [19] J. Hong, S. Lim and W. Lee, "Wavelength Converter Assignment Problem in All Optical WDM Networks," *Lecture Notes in Computer Science*, vol. 3681, pp. 1004-1013, 2005. [Article \(CrossRef Link\)](#)
- [20] C.S. Sung and J.M. Hong, "Branch-and-price algorithm for a multicast routing problem," *Journal of the Operational Research Society* 50, pp. 1168-1175, 1999. [Article \(CrossRef Link\)](#)
- [21] J. Hong and S. Lim, "Branch and Price Algorithm for Content Allocation Problem in VOD Network," *Lecture Notes in Computer Science*, vol. 3483, pp. 642-651, 2005. [Article \(CrossRef Link\)](#)
- [22] Monique Guignard, "Lagrangian Relaxation," *Top*, vol. 11, no. 2, pp. 151-228, 2003. [Article \(CrossRef Link\)](#)
- [23] D. Plassmann and N. Silva, "Mobility management functions for MBS," R2067/WP211/034.b1, Project Report, Aachen University of Technology, 1993.



**Jung Man Hong** received the B.S. degree in Industrial Engineering from Korea University, Seoul, Korea, in 1993 and the M.S. degree and Ph.D. in industrial engineering from Korea Advanced Institute of Science and Technology, Taejeon, in 1995 and 2000, respectively. Since 2002, he has been a Business Consultant at Entru Consulting Partners, LG CNS in Seoul. His research interests include wireless communication system analysis, integer programming and combinatorial optimization.



**Jong Hyup Lee** received the B.S. degree in industrial engineering in 1984 from Korea University in Seoul, Korea, and the M.S. and Ph.D. degrees in industrial engineering from KAIST in Korea, in 1986 and 1996, respectively. From 1986 to 2003, he worked for ETRI as a project leader. In ETRI, he was involved in research and development of TDX-1A/TDX-10 large-scale digital switching systems, ATM access nodes and 10Giga Ethernet switching systems, etc. Since 2004, he has served as an associate professor in the Department of Information and Communications Engineering at Inje University, Korea. His current research interests include the network planning and optimization, Internet QoS and mobility management, network design and routing for the advanced networks.



**Soong-Hee Lee** received the B.S., M.S., and Ph.D. degrees from Kyungpook National University, Daegu in 1987, 1990, and 1995, respectively. From 1987 to 1977, he was a member of research staff in Electronics and Telecommunications Research Institute. Since 1997, he has been with the Dep't of Information and Communications Engineering, Inje University as an associate professor. His research activities are in the area of next generation network technologies and services. He was designated as an IT Standard Expert on behalf of Korea by Ministry of Information and Communications in 2001. He is an Editor in the ITU-T SG13 and a Member of KSII.