Post-deployment Tuning of UMTS Cellular Networks through Dual-homing of RNCs

Samir K Sadhukhan
Indian Institute of Management (IIM)
Calcutta, India
samir@iimcal.ac.in

Swarup Mandal
Wipro Technologies
Kolkata, India
swarup.mandal@wipro.com

Saroj R Biswas
IBM Global,
Kolkata, India
saroj.biswas@in.ibm.com

Partha Bhaumik
CSE Dept, Jadavpur Univ, Kolkata, India
partha.bhaumik@gmail.com

Debashis Saha
Indian Institute of Management (IIM)
Calcutta, India
ds@iimcal.ac.in

***Abstract*— In conventional UMTS cellular networks, during deployment usually a set of NodeBs is assigned to one Radio Network Controller (RNC), and a set of RNCs to one Serving GPRS Support Node (SGSN) for data services, as well as to one Mobile Switching Centre (MSC) for voice services. Operators thus far have considered single-homing of RNCs to MSCs/SGSNs (i.e., many-to-one mapping) with an objective to reduce the total cost over a fixed period of time. However, a single-homing network does not remain cost-effective any more when subscribers later on begin to show specific inter-MSC/SGSN mobility patterns (say, diurnality of office goers) over time. This necessitates post-deployment topological extension of the network in terms of dual-homing of RNCs, in which some specific RNCs are connected to two MSCs/SGSNs via direct links resulting in a more complex many-to-two mapping structure in parts of the network. The partial dual-homing attempts to increase link cost minimally and reduce handoff cost maximally, thereby significantly reducing the total cost in a post-deployment optimal extension. In this paper, we formulate the scenario as a combinatorial optimization problem and solve the NP-Complete problem using two meta-heuristic techniques, namely Simulated Annealing (SA) and Tabu search (TS). We then compare these techniques with a novel optimal heuristic search method that we propose typically to solve the problem. The comparative results reveal that, though all of them perform equally well for small networks, for larger networks, the search-based method is more efficient than meta-heuristic techniques in finding optimal solutions quickly.**

***Keywords*- Network planning; Cellular network; UMTS; Dual-homing; Optimization; Simulated Annealing; Tabu Search; Heuristic search**

I. INTRODUCTION

Post-deployment planning plays a key role in optimizing the total cost of operation for modern cellular networks [1]-[5]. The dynamic nature of subscriber's profile makes the operation of cellular networks become sub-optimal with the passage of time in terms of the operational cost, and, hence, re-planning of networks needs to be done from time to time, with the existing deployment as a set of constraints (to protect investments).

An operator usually face one of the following three possible scenarios in post-deployment tuning phase: (i) both new call traffic and handoff traffic increase, (ii) new call traffic increases but handoff traffic does not, (iii) new call traffic does not increase but handoff traffic does. The first two cases usually occur when subscribers' density increases permanently. This can be addressed in post deployment planning phase by splitting cells (where capital expenditure as well as hand-off cost will increase) or by redefining the connectivity of cells and switches. The third case may arise due to a gradual change in mobility pattern of the existing subscriber base over a long period of time [6]. This problem can be addressed by regrouping cells into new clusters i.e., by changing the connectivity of NodeBs to RNCs and RNCs to MSCs/SGSNs [3]-[7].

However, this cannot take care of situations where handoff increases with no increase in total traffic due to periodic (temporal) changes of subscribers' locations. If there is a clear pattern of this temporal mobility of subscribers, a multi-homing consideration (where many NodeBs are connected to one RNC, and many RNCs are connected to one MSC/SGSN) will be a useful strategy in post deployment tuning stage. Obviously, the multi-homing concept can be implemented at two levels, namely, in the first level, multi-homing of NodeBs, and, in the second level, multi-homing of RNCs. In this paper, we have considered dual-homing of RNCs, where some RNCs (to be decided optimally) are connected to two MSCs/SGSNs (as shown in Fig. 1) to reduce handoff cost, unlike single homing where one RNC is connected to one MSC/SGSN only. In order to achieve an optimal selection of RNCs from the set of potential RNCs to be dual-homed, we have proposed three techniques in this work.

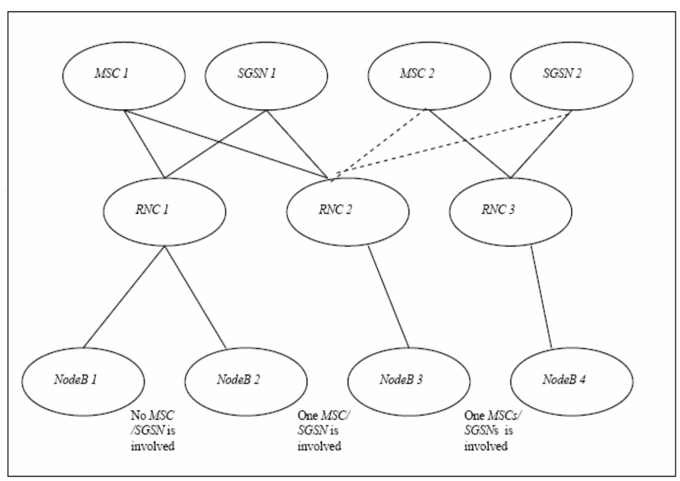


Figure 1. Change in handoff type due to dual-homing of RNC in a UMTS network (only relevant part of the UMTS network is shown; dotted lines indicate additional connections)

We define three types of handoff, namely simplest, simple and complex. We call a handoff simplest when no MSC/SGSN is involved in the handoff process. In case of simplest handoff, one RNC handles the whole handoff process. On the contrary, a simple handoff involves one (and only one) MSC/SGSN in the handoff process, whereas a complex handoff involves two MSCs/SGSNs. Obviously, simplest handoff cost is negligible with respect to complex handoff cost. Dual-homing of RNC is attractive for operators because it converts many complex handoffs to simple handoffs. For example, the com-plex handoff between NodeB 3 and NodeB 4 has changed to simple handoff due to dual-homing of RNC 2 in Fig. 1.

Researchers have traditionally formulated the NodeB-RNC (earlier known as cell-switch) assignment problem as a combinatorial optimization problem and have solved it using meta-heuristics [3]-[7] or domain-specific heuristics [8]-[13]. However, till date they all have considered single homing criteria, and have excluded the multi-homing scenario. Recently, we attempted to solve the dual homing of RNCs to MSCs in 2.5G networks with a suboptimal greedy algorithm [13] following the approach given in [14], which, however, deals with ATM networks (not cellular) and employs the meta-heuristic genetic algorithm (GA). The state space formulation of the problem was absent in [13], and, hence, we could not devise any domain-specific heuristic there. In the current work, we have extended our earlier work to UMTS networks for dual homing of RNCs to MSCs/SGSNs and solved it using three separate techniques. Firstly, we have used simulated annealing (SA) [3] and tabu search (TS) [5] meta-heuristic techniques and then developed a novel optimal search method with a polynomial time admissible heuristic [11],[16]. The heuristics method is reasonably strong that works fine with even larger networks, where SA and TS fail to give optimal solutions.

The paper is organized in six sections. Following Section I that introduces the dual homing assignment problem, Section II presents an integer linear programming (ILP) formulation of the problem. Section III and Section IV discuss the SA and TS based solution methodologies. In Section V, the optimal heuristic search algorithm is presented. Section VI contains the experimental results with discussion, and Section VII concludes the paper.

II. MATHEMATICAL FORMULATION

Let us consider that, in the UMTS network of a mobile telecom service provider (MTSP), there are n NodeBs, r RNCs, m MSCs and s SGSNs, whose locations are known. Let I ={1,2, … n} denote the set of NodeBs, J= {1,2, … r} denote the set of RNCs, K={1,2, …m} denote the set of MSCs and L={1,2, … s} denote the set of SGSNs. We assume that, from the existing single homing network, the initial assignments of NodeBs to RNCs and of RNCs to MSCs/SGSNs are known a priori. Throughout this formulation, we use a small letter to denote a member of the set represented by the corresponding capital letter; for example, $i \in I$, $j \in J$, $k \in K$, $l \in L$. Moreover, we assume NodeB i and NodeB i′ are different NodeBs ($i \neq i'$). Similarly we assume $j \neq j'$, $k \neq k'$, $l \neq l'$. Let us further define the following binary variables:

x'_{jk} = 1, if RNC j is assigned to MSC k (new link) in dual home network, 0, otherwise

x''_{jl} = 1, if RNC j is assigned to SGSN l (new link) in dual home network, 0, otherwise

d_{ij} = 1, if NodeB i is assigned to RNC j (old link) in single home network, 0, otherwise

d'_{jk} = 1, if RNC j is assigned to MSC k (old link) in single home network, 0, otherwise

d''_{jl} = 1, if RNC j is assigned to SGSN l (old link) in single home network, 0, otherwise

c'_{jk} is the amortization cost of the link between RNC j and MSC k

c''_{jl} is the amortization cost of the link between RNC j and SGSN l

Cap_j and Cap'_j are the capacity in circuit switching (number of calls per unit of time) and capacity in packet switching (bits per second) of RNC j.

Cap''_k and Cap'''_l are the capacity in circuit switching (number of calls per unit of time) and capacity in packet switching (bits per second) of MSC k and SGSN l respectively.

f^{voice}_{jk} is the amount of voice traffic produced by RNC j destined to MSC k

f^{data}_{jl} is the amount of data traffic produced by RNC j destined to SGSN l

$H^{msc}_{ii'}$ is the cost per unit of time for complex handoff between NodeB i and NodeB i′ involving two MSCs

$H^{sgsn}_{ii'}$ is the cost per unit of time for complex handoff between NodeB i and NodeB i′ involving two SGSNs

Next, let us define the following composite variables:

$$p_{ijk} = d_{ij} d'_{jk}, \; u_{ik} = \sum_{j \in J} p_{ijk} \quad \text{Then}$$

u_{ik} is equal to 1 if there is a path from NodeB i to MSC k

$$\hat{p}_{ijk} = d_{ij} x'_{jk}, \; \hat{u}_{ik} = \sum_{j \in J} \hat{p}_{ijk} \quad \text{Then}$$

$\hat{u}_{ik}$ is equal to 1 if there is a new path NodeB i to MSC k

Then $\hat{u}_{ik} \hat{u}_{i'k} \vee u_{ik} \hat{u}_{i'k} \vee \hat{u}_{ik} u_{i'k}$ is equal to 1 if NodeB i and NodeB i′ are under MSC k using at least one new path.

Moreover $\bigwedge_{k\in K}\left(\hat{u}_{ik}\hat{u}_{i'k} \vee u_{ik}\hat{u}_{i'k} \vee \hat{u}_{ik}u_{i'k}\right)$ is equal to 1 if *NodeB i* and *NodeB i′* are under one *MSC* using at least one new path.

Let us define the following variables:

$q_{ijl} = d_{ij}d''_{jl}$, $v_{il} = \sum_{j\in J} q_{ijl}$ Then

v_{il} is equal to 1 if there is a path from NodeB i to SGSN l

$\hat{q}_{ijl} = d_{ij}x''_{jl}$, $\hat{v}_{il} = \sum_{j\in J}\hat{q}_{ijl}$ Then

$\hat{v}_{il}$ is equal to 1 if there is a new path NodeB i to SGSN l

Then $\hat{v}_{il}\hat{v}_{i'l} \vee v_{il}\hat{v}_{i'l} \vee \hat{v}_{il}v_{i'l}$ is equal to 1 if NodeB i and NodeB i′ are under SGSN l using at least one new path.

Moreover $\bigwedge_{l\in L}\left(\hat{v}_{il}\hat{v}_{i'l} \vee v_{il}\hat{v}_{i'l} \vee \hat{v}_{il}v_{i'l}\right)$ is equal to 1 if NodeB i and NodeB i′ are under one SGSN using at least one new path.

Therefore the total reduction of complex handoff in the dual-home network will be

$$\sum_{i\in I}\sum_{i'\in I}\bigwedge_{k\in K}(\hat{u}_{ik}\hat{u}_{i'k} \vee u_{ik}\hat{u}_{i'k} \vee \hat{u}_{ik}u_{i'k})H^{msc}_{ii'}$$
$$+\sum_{i\in I}\sum_{i'\in I}\bigwedge_{l\in L}(\hat{v}_{il}\hat{v}_{i'l} \vee v_{il}\hat{v}_{i'l} \vee \hat{v}_{il}v_{i'l})H^{sgsn}_{ii'}$$

The total cost of the new links in the dual-home network will be

$$\sum_{j\in J}\sum_{k\in K}x'_{jk}c'_{jk} + \sum_{j\in J}\sum_{l\in L}x''_{jl}c''_{jl}$$

The link constraints are

$$x'_{jk} + d'_{jk} \le 1$$

The link constraint signifies that there could be at most one link (either an old link or a new link or no link) between RNC j and MSC k.

Similarly, $x''_{jl} + d''_{jl} \le 1$

$$\sum_{k\in K}x'_{jk} \le 1$$

The above constraint signifies that *RNC j* can be connected to at most one *MSC* using a new link.

Similarly, $\sum_{l\in L}x''_{jl} \le 1$

The capacity constraints can be considered in two ways:

(i) The worst case capacity constraints (when the capacity of RNCs are considered) are

$$\sum_{j\in J}cap_j\left(x'_{jk} + d'_{jk}\right) \le cap''_k$$

$$\sum_{j\in J}cap'_i\left(x''_{jl} + d''_{jl}\right) \le cap'''_l$$

(ii) The best case capacity constraints (when the capacity utilization of RNCs are considered) are

$$\sum_{j\in J}f^{voice}_{jk}\left(x'_{jk} + d'_{jk}\right) \le cap''_k$$

$$\sum_{j\in J}f^{data}_{jl}\left(x''_{jl} + d''_{jl}\right) \le cap'''_l$$

Therefore the dual-homing problem can be formulated as

Maximize

$$\sum_{i\in I}\sum_{i'\in I}\bigwedge_{k\in K}(\hat{u}_{ik}\hat{u}_{i'k} \vee u_{ik}\hat{u}_{i'k} \vee \hat{u}_{ik}u_{i'k})H^{msc}_{ii'} - \sum_{j\in J}\sum_{k\in K}x'_{jk}c'_{jk}$$
$$+\sum_{i\in I}\sum_{i'\in I}\bigwedge_{l\in L}(\hat{v}_{il}\hat{v}_{i'l} \vee v_{il}\hat{v}_{i'l} \vee \hat{v}_{il}v_{i'l})H^{sgsn}_{ii'} - \sum_{j\in J}\sum_{l\in L}x''_{jl}c''_{jl} \quad (i \ne i')$$

Subject to

$$p_{ijk} = d_{ij}d'_{jk} \quad i\in I, j\in J, k\in K \tag{1}$$

$$u_{ik} = \sum_{j\in J}p_{ijk} \quad i\in I, k\in K \tag{2}$$

$$\hat{p}_{ijk} = d_{ij}x'_{jk} \quad i\in I, j\in J, k\in K \tag{3}$$

$$\hat{u}_{ik} = \sum_{j\in J}\hat{p}_{ijk} \quad i\in I, k\in K \tag{4}$$

$$q_{ijl} = d_{ij}d''_{jl} \quad i\in I, j\in J, l\in L \tag{5}$$

$$v_{il} = \sum_{j\in J}q_{ijl} \quad i\in I, l\in L \tag{6}$$

$$\hat{q}_{ijl} = d_{ij}x''_{jl} \quad i\in I, j\in J, l\in L \tag{7}$$

$$\hat{v}_{il} = \sum_{j\in J}\hat{q}_{ijl} \quad i\in I, l\in L \tag{8}$$

$$x'_{jk} + d'_{jk} \le 1 \quad j\in J, k\in K \tag{9}$$

$$x''_{jl} + d''_{jl} \le 1 \quad j\in J, l\in L \tag{10}$$

$$\sum_{k\in K}x'_{jk} \le 1 \quad j\in J \tag{11}$$

$$\sum_{l\in L}x''_{jl} \le 1 \quad j\in J \tag{12}$$

$$\sum_{j\in J}cap_j\left(x'_{jk} + d'_{jk}\right) \le cap''_k \quad k\in K \tag{13}$$

$$\sum_{j\in J}cap'_i\left(x''_{jl} + d''_{jl}\right) \le cap'''_l \quad l\in L \tag{14}$$

The objective function, formulated above, is not linear and contains boolean operator ‘OR’ and product of binary variables. The boolean operator ‘OR’ can be converted to arithmetic addition as follows:

$X \vee Y = X + Y - XY$, where X and Y are binary variable.

The nonlinear term XY (i.e., the product of two binary variables) can be converted to linear form using additional variable $Z = XY$ (Z is binary too) and by adding the following type of constraints in the above formulation.

$$Z \leq X$$
$$Z \leq Y$$
$$Z \geq X + Y - 1$$

Similarly, product of more than two binary variables can be converted to linear form using additional binary variables and constraints. Thus, the above formulation of the problem can be converted to a 0-1 integer linear programming (ILP) problem which can be easily proven to be NP-Complete [17]. Thus, we can conclude that the problem of dual homing is NP-Complete. It is obvious from the formulation that an exhaustive enumeration technique for assigning RNCs in dual homing problem requires checking $(m+s)^r$ combinations. So, when the problem size is large, heuristic techniques are suitable to solve such problems [11].

A state S is defined by a set of connections, $S = \{C_1, C_2, C_3, C_4 \ldots\ldots\}$ represented in the form of a matrix. A connection C is an ordered pair <j-k> which implies that RNC j has been assigned (connected) to MSC k. The solution state space of the problem is the entire set of sets of connections generated by all possible and feasible combinations of <j-k> pairs. To illustrate the concept, let us show a toy UMTS network of three MSCs and six RNCs (NodeBs are omitted for the sake of simplicity) in Fig. 2. Spare capacities of MSCs are shown in square boxes next to them in Fig. 2.

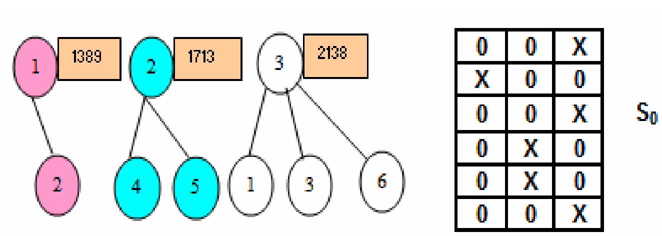


Figure 2. Initial single-home network and its state representation

S_0 (Fig. 2) represents the initial state for the network; we assume that a state matrix represents the network in terms of a connectivity matrix whose rows indicate RNCs and columns indicate MSCs. State representation of a network is explained in more details in Section V. Let us consider that S_0 is the initial single-homed network that needs post-deployment tuning now. We shall use this toy network in explaining our proposed algorithms in subsequent sections. Table 1 shows the capacities of RNCs, and Table 2 shows the handoff costs between RNCs for S_0. Table 3 shows the amortized cost of the high speed links from RNCs to MSCs.

TABLE 1. RNC capacity

RNC	1	2	3	4	5	6
Capacity	480	1158	318	1188	240	984

TABLE 2. Handoff Matrix for S_0

RNC \ RNC	1	2	3	4	5	6
1	-	-	-	9	14	-
2	-	-	19	4	-	24
3	-	19	-	5	6	-
4	9	4	5	-	-	-
5	14	-	6	-	-	-
6	-	24	-	-	-	-

TABLE 3. Link Cost

RNC \ MSC	1	2	3
1	36	36	-
2	-	36	36
3	20	26	17
4	36	-	36
5	26	17	20
6	35	44	10

III. SIMULATED ANNEALING TECHNIQUE

Simulated Annealing (SA) is a global optimization algorithm that can be applied to solve various combinatorial optimization problems. Annealing is a metallurgical process. It is basically heating followed by a slow cooling down process of a molten material in order to increase the crystal size and hence decrease their structural defects. Heat causes the atoms to get energized from their initial positions (a local minimum), wander randomly in states of higher energy, and then slow cooling gives the atoms chance to find configurations with lower internal energy than the initial state. SA-based techniques follow this approach for combinatorial optimization. By analogy with metallurgical annealing, SA randomly finds a neighbour of a current state, and then replaces the current state with this neighbour based on an acceptance probability which depends on the costs of the two states and the temperature of the system. When the temperature of the system is high (i.e., in the beginning), the probability of an ‘uphill’ move (i.e., acceptance of a worse solution) is more. This prevents the search process from being stuck at a local optimum. However, as the temperature of the system decreases slowly over time, the probability of accepting a worse solution decreases, and, at considerably low temperatures, almost all moves are ‘downhill’. When temperature becomes ‘zero’, the system freezes giving the lowest cost state as output.

The algorithm we have used here is not a pure SA algorithm. A pure SA algorithm does not keep track of the best solutions found in previous iterations. Thus, a pure algorithm, at the end of the iterations, always returns the current state (i.e., the state obtained at the last iteration) to be the solution. In contrast, our modified SA algorithm returns the best result of all the states visited during the iterations. We have found that, in several cases, this can be a significant improvement over the solution returned by the pure SA.

In SA, we start with an initial solution marked as the current node. Then, at each iteration, a child node is selected as the neighbour (the leftmost feasible child of the current node). We shall define neighbour below. If the acceptance probability of the selected node is greater than some threshold, the selected node is accepted for further exploration and is made the current node. If the selected node is not accepted, the next child is taken as the neighbour (i.e., the one to the right of the previous child). In this way the search continues. Typically, in this formulation, all children of a node are ‘neighbours’ of the node.

Neighbourhood Function: We define neighbourhood function of a state as a function which produces feasible children of the state. Neighbourhood generation of one example initial state S_1 (for a toy network of 4 RNCs and 3 MSCs) is shown in Fig. 3. All children of a state may not be feasible because some child states may correspond to infeasible network operation. We need to ignore them. So only the feasible children (without repetitions) of a parent state are its neighbour at the next level.

In the example shown in Fig. 3, S_2, S_3, S_4, S_5 and S_6 are children (neighbours) of the current initial state S_1. Neighbours of a node are generated by changing one connection of the node at a time, thus generating all permutations which differ from the parent state in exactly one RNC. In the shown example, S_2 differs from the parent at RNC 1, S_3 differs in RNC 2, S_4 differs in RNC 3, S_5 and S_6 differ from the parent at RNC 4 and so on. Hence, all pairs of nodes that differ from each other in exactly one RNC are neighbours of each other also.

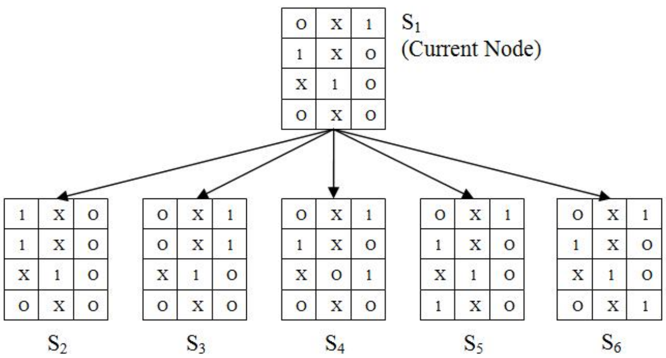


Figure 3. Neighbourhood nodes (states) of an initial node (state)

Reachability of Neighbourhood Function: The neighbourhood function we are using is exhaustive i.e., all possible combinations of RNC-to-MSC assignments are generated in this search tree. Hence, every node can be reached from every other node and the reachability criterion is satisfied.

Selection Criteria for new current state: If the cost C′ of any randomly chosen neighbour S′ of the current state S is less than the cost C of current state S, then the neighbour S′ is selected as the new current state for the next iteration. Otherwise, a random number R(0,1) between 0 and 1 is chosen from a uniform distribution. If P(C,C′,T) > R(0,1), the neighbor S′ is selected as the new current state (T is the current temperature, and function P is defined below). Otherwise, the present current state is propagated to the next iteration as the new current state. P(C,C′,T) is the acceptance probability and is defined by the Boltzmann probability factor as follows:

$$P(C,C',T) = \begin{cases} 1 & \text{, if } C' < C \\ \exp(-\Delta C/T) \quad \text{where } \Delta C = C'-C & \text{, otherwise} \end{cases}$$

Annealing Schedule: Typically, update of the temperate T is done by using the relation $T = T * \alpha$ (α is known as the cooling rate) after every V iterations, starting with the initial temperature $T = T_0$.

Thus, the various parameters for the SA algorithm are:

- α , the cooling rate i.e., the rate at which the temperature decreases ($0 < \alpha < 1$)
- T_0, the initial temperature
- V, the maximum number of iterations at a particular temperature
- γ, the maximum number of consecutive acceptance of worse solutions
- I, the maximum number of iterations, I >> V

Initial Feasible Solution: First an initial solution has to be generated which becomes the current state S for the very first iteration. To generate an initial feasible solution, we take the following heuristic approach (algorithm is given below). We maintain internally a table which stores the possible handoff reduction that can be achieved by connecting an RNC to an MSC. We connect that RNC to the MSC only if (i) the RNC is not already dual-homed, (ii) the reduction achieved is greater than the link cost, and (iii) the capacity of MSC is sufficient to accommodate the capacity of the RNC. Since we are designing a dual-homing approach, an RNC cannot have more than 2 links i.e., an RNC cannot be connected to more than two MSCs. After connecting an RNC to an MSC, the capacity of the MSC is reduced by an amount equal to the capacity of the RNC. We simply try to connect RNCs to MSCs one after another in this greedy fashion provided the capacity constraint and link constraint are satisfied for each new connection. This ultimately results in a complete solution after which no more connections are possible. This is then taken as the initial feasible solution. Then the main algorithm sets out to improving this initial solution.

<u>Algorithm GENERATE_INITIAL_SOLUTION:</u>

```
for k=0 to MAX_NO_OF_MSCS
{   for j=0 to MAX_NO_OF_RNCS
  {     if RNC j is connectable to MSC k ( i.e. link cost from
          RNC j to MSC k  < handoff cost  reduction
          & spare capacity of MSC k is  ≥ capacity of RNC j &
          number of links of RNC j is = 1)
        {  connect j to k;
           decrement the spare capacity of MSC k by the
                   capacity of RNC j;
           increment the number of links of RNC j by 1;
            update the tables that store info about the handoff
                   cost and handoff cost reduction;
        }
    }
}
```

The cost of the initial solution is also computed. The initial solution becomes the current solution for the first iteration. The iteration steps are performed in the main annealing process.

Example: For the network of Fig. 2, we take SA parameters as T_0=3000, V=50, α=0.5, I=1000. Link cost of single home network = 17+17+10 = 44. Total handoff cost of single home network = 9+14+19+4+24+19+5+6+9+4+5+ 14+6+24 = 162. Total cost of single home network = Link cost + Handoff cost = 44+162 = 206. The matrix representation of the toy single home network is also given in Fig. 2 as S_0. To generate the initial solution, connections are attempted one after another in a directed manner i.e., in the order 1-to-1, 2-to-1, 3-to-1, 4-to-1, 5-to-1, 6-to-1, 1-to-2, 2-to-2, 3-to-2 and so on. The connections already present in S_0 are ignored. If a connection is found to be feasible (i.e., capacity and link constraints are satisfied) then it is included. The initial solution thus generated and the RNC-RNC handoff costs for that solution are given in Fig. 4.

MSC / RNC	1	2	3
1	0	1	X
2	X	0	0
3	1	0	X
4	0	X	0
5	0	X	0
6	1	0	X

RNC / RNC	1	2	3	4	5	6
1	-	-	-	9	14	-
2	-	-	-	4		-
3	-	-	-	5	6	-
4	9	4	5	-	-	-
5	14		6	-	-	-
6	-	-	-	-	-	-

Figure 4. (a) Intial solution (b) Handoff costs

Total cost of this solution = Single home link cost + Dual home link cost + Handoff cost=44+ (36+20+35) + (4+5+6+ 4+5+6) = 165. Hence, cost reduction = Single home cost – dual home cost = 206 – 165 = 41.

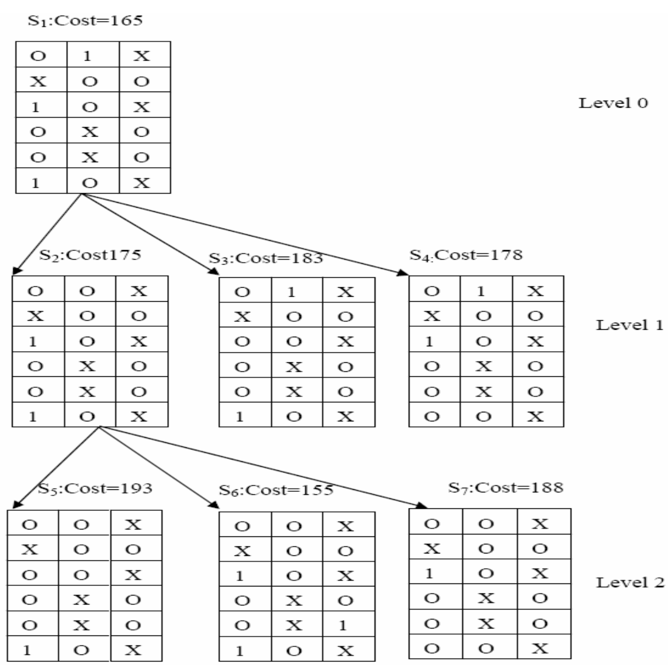


Figure 5. Neighbour generation. Only feasible nodes are shown.

Now, taking this initial solution as the root node (current node), the tree traversal starts in a breadth first manner. Children are accepted probabilistically and new solutions are checked for improvements over the previously found best solution. The leftmost feasible child of the current node and the corresponding handoff matrix are shown in Fig. 6.

Cost of this child (neighbour) = Single home link cost + Dual home link cost + Handoff cost = 44+(20+35)+(4+5+6+ 9+14+4+5+6+9+14) = 175. Hence, cost reduction=Single home cost – dual home cost = 206 – 175 = 31. Also, acceptance probability = exp (-ΔC/T) = exp (- (175-165)/3000) = 0.996672. Thus, at initial high temperature, we see the probability of accepting a worse solution is very high, and we accept it. The tree traversal continues in this fashion, and each child is checked for acceptance. After every 50 iterations, the temperature of the system is halved (α=0.5). We stop when the iteration count reaches the a priori set value of I.

MSC / RNC	1	2	3
1	0	0	X
2	X	0	0
3	1	0	X
4	0	X	0
5	0	X	0
6	1	0	X

RNC / RNC	1	2	3	4	5	6
1	-	-	-	-	-	-
2	-	-	-	4		-
3	-	-	-	5	6	-
4	-	4	5	-	-	-
5	-		6	-	-	-
6	-	-	-	-	-	-

Figure 6. (a) New solution (b) Handoff costs

IV. TABU SEARCH TECHNIQUE

This process is very similar to SA. The word "tabu" describes a sacred place or object. Things that are tabu must be left alone and should not be visited or touched. Tabu Search extends hill climbing by this concept– it declares solution candidates which have already been visited as tabu. Hence, they must not be visited again, and the optimization process is less likely to get stuck to a local optimum. The simplest realization of this approach is to use a list of tabu, which stores all solution states that have already been visited. If a newly generated neighbour can be found in this list, it is not accepted but rejected right away. Of course, the list cannot grow infinitely but has a finite maximum length N.

The tabu search (TS) algorithm we are using is straightforward. We start with an initial solution as current solution (say S) which can be found randomly. In every iteration of TS, we find a new solution by making local movements on this current solution. The neighbourhood function is used to find all neighbours of the current state S i.e., the neighbourhood set of S. The next solution state is the best solution among all the possible neighbours of S which are not already in the tabu list. Here, we use the same state space formulation and neighbourhood function as those used in SA.

Typically there are two kinds of tabu lists- (i) a long term memory maintaining the history through all the exploration process as a whole, and (ii) a short term memory to keep the most recently visited tabu movements. The first approach can actually produce the optimal solution if the state space search algorithm is exhaustive and the reachability criterion is satisfied. We use the second approach for faster solution. The tabu list has a finite length which is a parameter for the process. Whenever a case arises in which we have to add a solution to the tabu list and the list is full, we first remove the oldest solution from the list and then add the new one.

Neighbourhood Function: The neighbourhood generation used here is similar to that used in SA (Fig. 3). However, unlike a single neighbour in SA, all neighbours of the current node are generated. Then, the best child (neighbour), not already in the tabu list, is selected as the current node and search continues in the same manner. The various parameters needed for the TS algorithm are:

- N, the length of the tabu list i.e., how many recent solutions are to be remembered
- I, the maximum number of iterations

Initial Feasible Solution: Same approach is used as it is in the case of SA.

Example: For the network of Fig. 2, we take parameters I=3, N=2 (size of tabu list). Initial solution is generated similar to the one found in SA and hence is as given in Fig. 4. Total cost of this solution is 165 and cost reduction is 41. This solution (S_1) is put into the tabu list as the first entry. As shown in Fig. 5, the initial solution is made as the root node. Of the three feasible children in level 1, the costs (from left to right respectively) are 175, 183 and 178. Hence, the best child is the leftmost child (S_2), and it is not in the tabu list. It is accepted, made the current node (to be explored next) and entered into the tabu list. The list now contains S_1 and S_2. At level 2 (iteration 2), there are three feasible children. The costs are 193, 155 and 188 respectively (from left to right). Of these, the best child is accepted i.e., S_6, the middle one (cost=155) which is not already in the tabu list and entered into the tabu list. The tabu list now contains S_2 and S_6. S_6 is then made the current node and search continues in this way.

V. STATE SPACE FORMULATION AND OPTIMAL HEURISTIC SEARCH ALGORITHM

The concept behind this algorithm is similar to our previous work [11]. It is based on a state space representation of the problem [16] (as exemplified in Fig. 7). For example, in this case, each state S (Fig. 7) has four rows and three columns where row *i* corresponds to RNC_i and column *j* corresponds to $MSC_j/SGSN_j$. The *s(i,j)* is an element of the state S where *s(i,j)* assumes a value 0 when RNC_i is not connected to $MSC_j/SGSN_j$, and 1 means the RNC_i is connected (i.e., dual homed) to $MSCj/SGSN_j$ and assumes *'x'* when the RNC_i is connected to MSC_j (in the single homed network). The root state provides information on links in the existing single homed network. For *r RNCs* and *m MSCs*, the root state has *(m-1)*r* child states (level 1), each state at level 1 has *(m-1)*(r-1)* child states, each state at level 3 has *(m-1)*(r-2)* child states and so on. A state at a level *i* has *(m-1)*(r-i+1)* child states [$i \leq r$]. There are n levels in the state space.

The complete state space contains $\sum_{i=0}^{r} (m-1)^i \, {}^{r}P_i$ states where *P* stands for permutation of numbers. In single homed network, call handling capacities of each *MSCs* and *RNCs* are known. The existing handoff costs between *RNCs* and the amortize link costs from each *RNCs* to each *MSCs* are known. The objective is to reduce the handoff cost of the network making *RNCs* dual-homed. An *RNC* can be connected to an *MSC* (for dual homing), if the call handling capacity of the *MSC* is not less than the call handling capacity of the *RNC* (capacity constraint) and the amortized link cost between the *RNC* and the *MSC* is less than the handoff cost reduction that arises due to dual homing of the *RNC* (cost constraint). We have presented an admissible heuristic function *h(S)* of total possible handoff reduction for a state *S* and Optimal Dual Home (ODH) Algorithm considering a voice network (network consists of *NodeBs*, *RNCs* and *MSCs*) which is used for finding optimal solution for the problem. Similar heuristic and algorithm can be formulated for data network as well as a combination of voice and data networks.

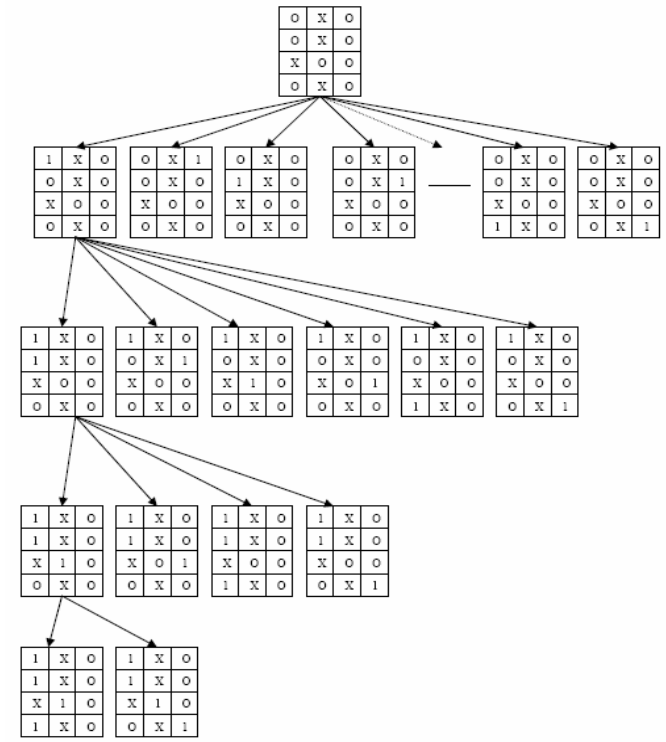


Figure 7. Dual home State Space

<u>*Algorithm ODH:*</u>

Step-1: INITIALIZATION:

I. Initialize the search process with a root state *r*, where *r* represents the single homing configuration of the RNCs-MSCs connectivity.
II. Initialize the cost of solution, *COST* as sum of cable cost and handoff cost. Set *S_COST = COST, RED=0*.
III. Initialize the solution *G* as *r*.
IV. PUSH *r* in the stack *STK*.

Step-2: EXPANSION:

I. **IF** *STK* is empty then GOTO Step-3
ELSE POP a state *S* from *STK*. **IF** *h(S) < RED* GOTO Step-2.I **ELSE** find all feasible child states (assignment of RNCs which satisfy capacity constraint and cost constraint) of *S*.
II. **IF** no feasible child state is found then mark the POPPED state *S* as a solution and find the cost of *S*.
IF cost of *S* is less than *COST*
  i. Update *COST* with cost of *S*
  ii. Update *G* as *S*.
  iii. RED = S_COST - COST
  iv. GOTO Step-2.I
ELSE GOTO Step-2.I
ELSE find the heuristic values for each of the feasible child states.

III. PUSH the feasible child states in *STK* in ascending order of heuristic values.
IV. GOTO Step-2.I

Step-3: TERMINATION:
Output *S* (the dual-homed configuration) and solution cost, *COST* (the dual-homed network cost).□

Calculation of h(S):

Let $scap(MSC_i)$ be spare call handling capacity of MSC_i $(i=1,2 \dots m)$. Let $cap(RNC_j)$ be call handling capacity of RNC_j $(j=1,2,\dots r)$. Let $L(RNC_j,MSC_i)$ be amortize link cost between RNC_j and MSC_i. Let $H[r,r]$ be the RNC-RNC handoff matrix. Here *m* is the number of *MSC* and *r* is the number of *RNC*.

Step-1: INITIALIZATION:
I. Initialize the heuristic value, *h(S)*, to zero.
II. Initialize the array of size *r*, $H_{Max\_Red\_MSC}[m]$ to zero (to store maximum possible handoff reduction for each MSC).
III. Set i = 1

Step-2: COMPUTATION:
I. **IF** $i > m$, GOTO Step-3 **ELSE** consider MSC_i
II. PUSH the single-homed RNCs which are not connected to MSC_i and satisfy the following conditions, in the stack *STK*.
   i. (capacity constraint) Capacity of RNC is less than $scap(MSC_i)$
   ii. (cost constraint) The amortized link cost between the RNC and the MSC_i is less than the handoff cost reduction that arises due to dual homing of the RNC
III. **IF** *STK* is empty then GOTO Step-2.VI **ELSE** POP RNC_j from *STK* and compute maximum possible handoff reduction for RNC_j i.e. $H^{RNCj}_{Red} = \sum_k(H[k,j] + H[j,k]) - \sum_p(H[p,j] + H[j,p])$, where $k = 1$ to r. Summation over *p* will be taken for $p = 1$ to r where RNC_p is dual homed and it is not connected to MSC_i.
IV. **IF** $H_{Max\_Red\_MSC}[i] < (scap(MSC_i)* H^{RNCj}_{Red} / cap(RNC_j) - L(RNC_j, MSC_i))$ then set $H_{Max\_Red\_MSC}[i] = scap(MSC_i)* H^{RNCj}_{Red} / cap(RNC_j) - L(RNC_j, MSC_i)$
V. GOTO Step-2.III.
VI. Compute $h(S) = h(S) + H_{Max\_Red\_MSC}[i]$. Set i = i + 1 and GOTO Step-2.I

Step-3: TERMINATION:
Output *h(S)*, the heuristic value of the state *S*. □

Theorem 1: *h(S)* is admissible heuristic and can be computed in $O(m*r)$.
Theorem 2: The algorithm ODH outputs optimal solution.

Proofs of the theorems are omitted due to space constraint.

Now the execution of ODH of the example network (Fig. 2) is shown below:

<u>Step-1: INITIALIZATION:</u>
I. Initialize the root state r = S_0.
II. Calculate cost of S_0,
COST = Total Handoff Cost + Total Cable Cost
= Sum of Handoff Matrix + Sum of Cable Cost for all RNC to MSC connection
= 162 + 44 = 206
S_COST = 206
III. Set G = S_0 and RED = 0
IV. PUSH the state S_0 in the stack STK.

<u>Step-2: EXPANSION:</u>
I. The stack STK is not empty. POP the state S_0 from STK. Since $h(S_0) \geq 0$, find all feasible child states of S_0.

<u>Feasible Child States Creation from S_0</u>

From the 1st row of S_0, it is found that RNC_1 can be made dual homed in two ways and hence two child states are generated (first two child states in Fig. 8). Similarly from the remaining rows of S_0, ten more child states are generated. Each child state provides one more new link from its parent state (S_0). Now the capacity constraint and cost constraint are checked for each child state. The feasible child states are those child states which satisfy both capacity constraint and cost constraint. For each feasible child state an updated RNC-RNC handoff matrix is maintained.

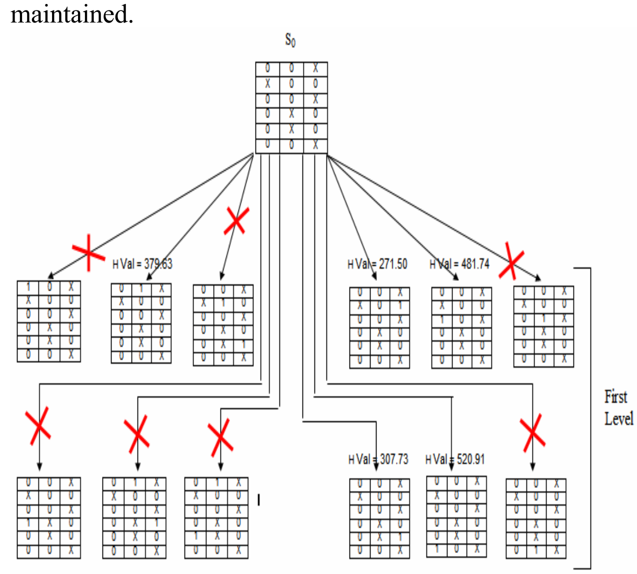
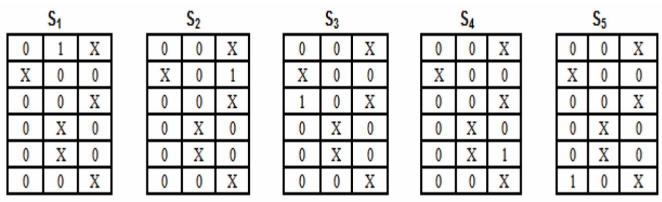


Figure 8. First Level Child States of S_0

Consider the first child state where RNC_1 is dual homed and connected to MSC_1. Here capacity constraint $scap(MSC_1) \geq cap(RNC_1)$ is satisfied as 1389 > 480. To check the cost constraint, we need to calculate total handoff reduction of the state. Total Handoff reduction = Handoff between RNC_1 and other RNCs that are connected to MSC_1 = $H(RNC_1, RNC_2) + H(RNC_2, RNC_1) = 0$. Link Cost from RNC_1 to MSC_1 is 36. So, the cost constraint is not satisfied as total handoff reduction < link cost between RNC_1 and MSC_1. Hence the first child is not a feasible child state. From Fig. 8, it is found that the following five child states of the S_0 are feasible child state.



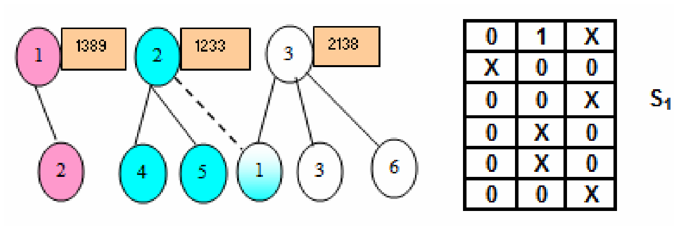


Figure 9. State S_1 of dual-home Network

TABLE 4: RNC-RNC Handoff Matrix for S_1

RNC \ RNC	1	2	3	4	5	6
1	-	-	-	0	0	-
2	-	-	19	4	-	24
3	-	19	-	5	6	-
4	0	4	5	-	-	-
5	0	-	6	-	-	-
6	-	24	-	-	-	-

Fig. 9 shows the dual-home network corresponding to the state S_1. The spare capacities of MSCs are shown in square boxes. To calculate $h(S_1)$ for state S_1, we need to compute maximum possible handoff reduction for all individual MSCs of S_1 and then sum them up. Let us first take MSC_1. RNC_1 is already dual homed and RNC_2 is connected to MSC_1. Hence, we need not consider RNC_1 and RNC_2 further. Here, $s(3,1) = 0$. Capacity of RNC_3 is 318 which is less than the spare capacity of MSC_1 (i.e., 1389). From Table 3, it is found that the amortized link cost between the RNC_3 and the MSC_1 is 20 which is less than the handoff cost reduction that arises due to dual homing of the RNC_3 i.e., $H(RNC_3, RNC_2) + H(RNC_2, RNC_3) = 38$. Similarly, from the Table 4, it is found that total possible handoff reduction for RNC_3 is 60 i.e., (19+19+5+5+6+6). From the first column of S_1, it is found that $s(4,1) = 0$, $s(5,1) = 0$ and $s(6,1) = 0$. But the RNC_4, RNC_5 and RNC_6 are not dual homed. Hence, value of H^{RNC3}_{Red} with respect to MSC_1 is 60. Maximum possible handoff reduction for RNC_3 = $scap(MSC_1)^* H^{RNC3}_{Red}$ / $cap(RNC_3)$ – $L(RNC_3, MSC_1)$ = {(1389*60/318) – 20} = 242.08. Similarly, value of maximum possible handoff reduction for RNC_4, RNC_5 and RNC_6 with respect to MSC_1 would be 0, 0 and 32.76, respectively. So, Maximum Possible Handoff reduction correspond to MSC_1 = Max {242.08, 0, 0, 32.76} = 242.08. Similarly, we calculate the maximum possible handoff reductions with respect to the remaining MSCs. Finally, we get $h(S_1)$= Sum of Maximum Possible Handoff reduction for all MSCs = 242.08 + 0 + 137.55 = 379.63. Similarly, we can calculate $h(S_2)$, $h(S_3)$, $h(S_4)$ and $h(S_5)$.

State	S_1	S_2	S_3	S_4	S_5
Heuristic value	379.63	271.50	481.74	307.73	520.91

III PUSH the feasible child states in STK in ascending order of its heuristic value.

STK =		S_5	S_3	S_1	S_4	S_2

IV. GOTO Step2.I

Continue to execute the above steps following the ODH algorithm till the stack STK gets emptied. For the single homed state S_0, it is found that the optimal dual home solution state $S_{optimal}$ (Fig. 10) is achieved using ODH algorithm. The optimal solution cost is 136 and total reduction of cost is = 206 – 136 = 70.

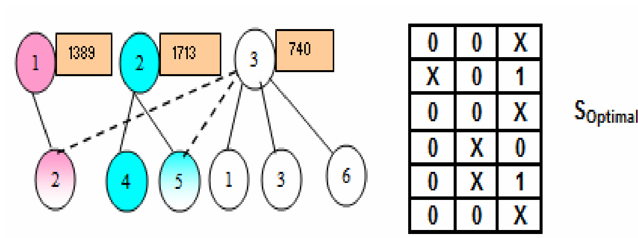


Figure 10. Optimal Dual-home Network and its state representation, $S_{optimal}$

VI. RESULTS AND DISCUSSION

We divide a rectangular area into multiple hexagonal NodeBs created using a non orthogonal Cartesian system inclined at 600. Each NodeB has exactly six neighbouring NodeBs except the boundary NodeBs that have less than six neighbouring NodeBs. A specified set of RNCs and MSCs are placed randomly in some NodeBs such that an MSC is co-located with an RNC. Subsequently, NodeBs are assigned to RNCs, and RNCs are assigned to MSCs using nearest neighbour distance.

After the creation of the above synthetic UMTS single home network architecture, voice handoff costs for neighbouring NodeBs are generated from a uniform random distribution. Amortized cost for a link from NodeB to RNC and RNC to MSC is taken proportional to the physical distance. Complex voice handoff costs between RNCs are calculated based on handoff cost between NodeBs. Spare capacities of MSCs and capacities of RNCs of the network are generated using capacities of NodeBs which are generated from a uniform distribution.

Algorithms SA, TS and ODH are run for 100 instances, and average solution cost is recorded against the triplet (MSCs, RNCs and NodeBs). Fig. 11 compares the algorithms with respect to average solution costs. The initial cost is taken as the cost of the single home network. For each dual-homed architecture solution, the reduction in handoff cost for the network is calculated, considering the cost of amortized link, subject to the capacity constraints of the MSCs and the RNCs.

The optimal dual-home network architecture and the corresponding cost of the network are found out from ODH. For TS, we have taken the tabu list size is equal to 10 and maximum number of iteration is equal to 1000. Average solution costs obtained by TS in most cases are close to optimal, when problem sizes are small. Fig. 12 shows that SA and TS increasingly fails to reach optimal solution cost with increase in problem size.

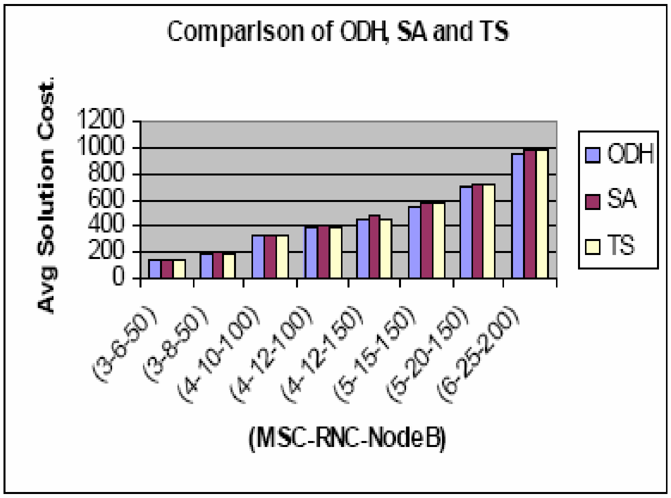


Figure 11. Dual-home solution costs

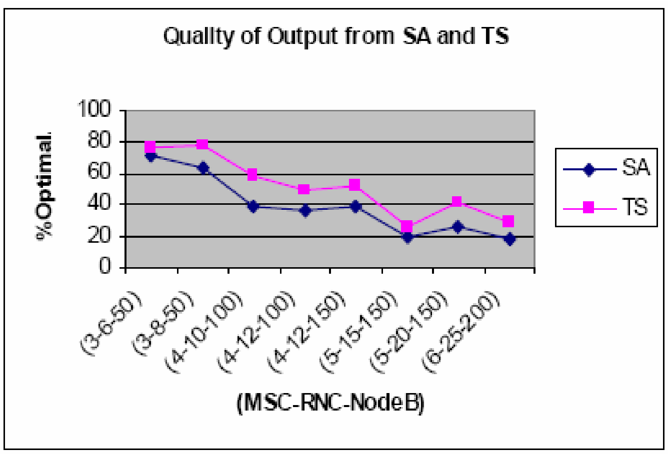


Figure 12. Percentage optimal output

VII. CONCLUSION

We have formulated the selective dual homing scenario of RNCs for post-deployment tuning of an existing single home UMTS networks as an ILP problem. Dual-homing of RNCs reduces the handoff cost but additional capacities are required at the MSCs/SGSNs in the network. The problem is difficult to solve in polynomial time as it falls into NP-Complete category. So we have first solved the problem using SA and TS (meta-heuristic) algorithms. We have then mapped the dual-homing problem into a state space and proposed an optimal heuristic search algorithm ODH of polynomial time heuristic $O(n^2)$ for the state space so formulated. It is found that TS technique is capable of finding good quality solutions which are better than those obtained by SA. But, when the problem size is large, the optimal solution from SA and TS becomes less likely, whereas the ODH can still produce optimal solutions. The technique will be helpful to the operators because the proposed solution can be implemented with minor changes in the protocol stack of the existing standards. We are currently working in this area.

REFERENCES

[1] P. Demestichas et. al., "Control of the location update and paging signaling load in cellular systems by means of planning tools", Proc. Fall. IEEE VTC, Vol. 4, pp: 2119 -2123, 1999.

[2] C. U. Saraydar, O.E. Kelly, C. Rose, "One-dimensional location area design", IEEE Transactions on Vehicular Technology, Vol. 49, Issue: 5, pp: 1626 – 1632, 2000.

[3] I. Demrikol, C. Ersoy, M. U. Caglayan, H. Delic, "Location area planning in cellular networks using Simulated Annealing", Proc. IEEE INFOCOM, 2001.

[4] A. Quintero, S. Pierre, "A Memetic algorithm for assigning cells to switches in cellular mobile networks", IEEE Communications Letters, Vol. 6, Issue:11, pp: 484 –486, 2002.

[5] S. Pierre, F Houeto, "Assigning cells to switches in cellular mobile networks using Taboo Search", IEEE Transactions on Systems, Man and Cybernetics, Part B, Vol. 32 , Issue: 3, pp:351 – 356, 2002.

[6] E. Clayirci, I. F. Akyildiz, "User mobility pattern scheme for location update and paging in wireless systems", Mobile Computing, IEEE Transactions,Vol.1 , Issue: 3 , pp:236 – 247, 2002.

[7] M. M. Diallo, S. Pierre, R. Beaubrun "An Approach for Cell Assignment in UMTS Networks" VTC, pp: 1-5, 2006.

[8] A. Merchant, B. Sengupta,"Assignment of cells to switches in PCS networks", IEEE/ACM Trans. on Networking, Vol.3,pp:521–526, 1995.

[9] D. Saha, A. Mukherjee, P. S. Bhattacharya, "A simple heuristic for assignment of cells to switches in a PCS network", Proc. Wireless Personal Communications, pp: 209- 224, 2000.

[10] P. S. Bhattacharjee, D. Saha, A. Mukherjee, "An approach for location area planning in a Personal Communication Services Network (PCSN)", IEEE Trans on Wireless Communication, 2004

[11] S. Mandal, D. Saha, and A. Mahanti, "Heuristic search techniques for cell to switch assignment in location area planning for cellular networks", Proc. IEEE International Conference on Communications (ICC), pp: 4307-4311, 2004.

[12] S. Mandal, D. Saha, and A. Mahanti, "A technique to support dynamic pricing strategy for differented cellular mobile services", Proc. IEEE Globecom., pp: 3388-3392, 2005

[13] S.. K. Sadhukhan, S. Mandal, and D. Saha, "A heuristic technique for solving dual-homing assignment problem of 2.5G cellular networks", Proc. IEEE International Conference on Computing: Theory and Applications, pp: 66-71, 2007

[14] D. R. Din, S. S. Tseng, 2005, "A genetic algorithm for solving dual-homing cell assignment problem of the two-level wireless ATM network", Computer Communications, 25 , 1536–1547, 2002

[15] D. Amzallag, J. Naor, D. Raz, "Algorithmic aspects of access networks design in B3g/4g cellular networks", Proc. IEEE INFOCOM, pp:991-999, 2007.

[16] J. Pearl, Heuristics: Intelligent search strategies for computer problem solving, Addison-Wesley, 1984.

[17] M. R. Garey, D. S. Johnson. Computers and Intractability: A guide to the theory of NP-Completeness, 1979.