Genetic Algorithms Applied to Cellular Call Admission Problem: Local Policies*

A. Yener & C. Rose
WINLAB
Dept. of Electrical and Computer Engineering
Rutgers University PO Box 909 Piscataway NJ 08855

Abstract

It is well known that if a stochastic service system (such as a cellular network) is shared by users with different characteristics (such as differing handoff rates or call holding times), the *overall* system performance can be improved by denial of service requests even when the excess capacity exists. Such selective denial of service based on system state is defined as call admission. A recent paper suggested the use of Genetic Algorithms to find near-optimal call admission policies for cellular networks. In this paper, we define *local* call admission policies that make admission decisions based on partial state information. We search for the best local call admission policies for one dimensional cellular networks using Genetic Algorithms and show that the performance of the best local policies is comparable to optima for small systems. We test our algorithm on larger systems and show that the local policies found outperform the maximum packing and best handoff reservation policies for the systems we have considered. We find that the local policies suggested by the Genetic Algorithm search in these cases are double threshold policies. We then find the best double threshold policies by exhaustive search for both one dimensional and Manhattan model cellular networks and show that they almost always outperform the best trunk reservation policies for these systems.

1 Introduction

Allocating radio resources to users with minimum blocking of new calls and dropping of handoffs has become a vital issue in mobile communication system design. It is known that dynamic channel allocation schemes can reduce these blocking and dropping probabilities for a reasonable range of loadings [1][2][3]. However, allocating channels to every user whenever the sources are available may not be the optimal strategy in terms of system performance. The performance can be improved (or equivalently the blocking probabilities can be further reduced) by imposing a state-based call admission policy on a system where there are users with differing service characteristics. The problem then becomes one of finding the call admission policy that provides optimal system performance.

The call admission problem has been studied extensively in the context of multirate circuit switching in [4] [5] [6] [7] [8]. Call admission to a single cell has been considered in [9]. In all cases, optimal policy search assumes an underlying Markov Process model and uses Dynamic Programming (Markov Decision Processes-MDP) to find the optimal policy. Unfortunately, issues such as state-space size often render MDP impractical.

In recent work [10] we used Genetic Algorithms (GA) to find near-optimal call admission policies in cellular networks to overcome the computational limits of MDP approach. In this paper, we extend the approach by considering call admission with incomplete state information. Specifically, we consider admission policies using only *local* rather than global channel occupancy information. The optimization problem based on local state is no longer Markovian and the MDP approach cannot be used directly.

Section 2 describes the network and the optimization problem. Section 3 defines local state-based policies and is followed by Section 4 which considers the application of GA to the local call admission problem. In section 5, we first test the algorithm on a small system and compare the performance to optima located using MDP. We then apply GA to a larger system and compare our results to well-known allocation techniques such as maximum packing and handoff reservation policies. We find that the GA found policies suggest a particular structure (a double threshold policy) and then we compare the best policies with this structure to maximum packing and reservation for both one dimensional and two dimensional (Manhattan model) cellular networks.

*Supported by NSF grant NCR-92-06148

2 Problem Definition

In this paper, we look into two types of cellular networks: a one dimensional ring structured network with an even number of cells and a two dimensional Manhattan model cellular network. We consider a system with two types of service requests, i.e. new call set up request and handoff request. We assume a Markov model in which the new call arrival process to every cell is modeled as Poisson with rate λ. Call holding times are exponential with average call completion rate μ. We also assume that calls in progress are subject to handoff to either of the two (or four for the two dimensional system) neighboring cells, and the time a mobile user spends in any cell is exponential with rate γ (independent of the call arrival and call holding process).

We define the cellular system performance, T, as a linear combination of call dropping and call blocking;

$$T = P_b + \omega P_h \quad (1)$$

where P_b and P_h are the new call blocking and handoff dropping probabilities respectively, and ω is the *relative penalty factor*. This performance measure penalizes dropping of a handoff $\omega > 1$ times more than blocking a new call request; an intuitively reasonable assumption. The reader should note that with $\omega = 1$, the performance measure is simply the probability of unsuccessful call completion,

$$T = P_b + P_h = 1 - P_c \quad (2)$$

where P_c is the probability of successful call completion. P_c is defined as the probability of not being blocked or dropped over the lifetime of a call from setup request to natural call termination.

3 Local Policies

A call admission policy is a collection of admit/reject decisions corresponding to the services requested at each state of the system. It can be described as a binary mapping where the admit decisions are represented by 1's and rejections by 0's. These admit/reject decisions are assumed to be independent of the system constraints, i.e. although a policy may try to admit a service request when the system is full, service will be denied due to unavailability. This formulation circumvents to problem of policy feasibility since *all* policies are then feasible.

We define a *global* call admission policy as one which provides a decision for each state of the network. For instance, a global policy for the one dimensional cellular system described in Section 2 provides the following decisions per cell per state:

- Admit/Reject a new call arrival
- Admit/Reject a handoff request from the right cell
- Admit/Reject a handoff request from the left cell

The size of the global call admission policy is then $3 \times N \times S$ bits, where N is the number of cells in the network and S is the number of feasible states of the system.

For practical systems, this size can be very large and may prohibit the use of MDP to find the optimal call admission policy. As an example, consider a 16-cell, 9-channel system. Since the number of states in the system is on the order of [1] $(C+1)^N$, the size of the policy is roughly 48×10^{16} bits. Furthermore, a global policy requires complete and up-to-date state information to be distributed to each cell. This may cause unacceptable signaling overhead for an already burdened signaling system[11].

An alternative approach is to define *local call admission policies* [12] where decisions are based only on partial state information. In this case, a cell (base station) would only keep track of the state information from small number of cells and make decisions based on the abbreviated state information which we define as the *local state*. Policies defined on the local state are therefore much smaller. This in turn means the policy search space is much smaller and policy search is simplified.

Definition 1 *A local policy is said to have* locality k *if it uses the state information from its* $2k$ *nearest neighbors.*

Thus, for the one dimensional system, a base station operating with a policy of locality k uses the state information from its k left and k right nearest neighbors as well as its own state. To further reduce policy complexity, we will assume a ring network for the one dimensional case and exploit its symmetry; i.e., the policy in each cell is assumed identical to that in other cells (within a spatial shift) [13]. The maximum possible state space then reduces to $(C+1)^{2k+1}$ states, and the length of the local policy becomes $3 \times (C+1)^{2k+1}$ bits per cell.

The locality conditions considered in this paper are $k = 0$ and $k = 1$. $k = 0$ corresponds to completely ignoring the state of adjacent cells whereas $k = 1$ corresponds to a local policy which uses the nearest neighbor state information in addition to self-state information. The above example with 16 cells and 9 channels needs only 30 bits for the policy with $k = 0$, and 3000 bits for $k = 1$.

It is worthwhile to restate that the search for optimal local policies is not amenable to MDP since the underlying state is no longer Markovian. Thus, Genetic Algorithms afford a systematic approach to a problem intractable using standard analytic methods. GAs perform surprisingly well in a variety of optimization tasks [14].

[1] The given number is the number of states in the unconstrained state space, the number of feasible states is actually less due to reuse constraints.

4 Genetic Algorithm Implementation

We use a typical two parent - two offspring Genetic Algorithm described in [10]. Each local policy represents an organism in GA terminology and is a collection of bits corresponding to admit (1) and reject (0) decisions for the new call set up and handoff requests at each *local* state of the system. A group of local policies (a community) is chosen at random initially. Each policy in the community is evaluated (via Monte Carlo simulation) using Equation 1 as the performance measure. Policies with better performance are more likely to enter the *mating pool*. These so-called *parent* organisms perform crossover to exchange some bit information pairwise and result in offspring policies. Finally, offspring policy bits are inverted randomly (mutation).

After mating, policies are chosen for deletion with probability inversely related to their fitness. The offspring are then inserted into the population. In our implementation, the population of the community is kept constant from iteration to iteration; we only delete as many policies as there are offspring. This basic algorithm is repeated for some number of *generations* or until policy improvement appears to stagnate.

We use Monte-Carlo simulation to evaluate dropping and blocking. In one dimensional case, we assume channel reuse such that a channel can only be used once in any K mutually interfering cells (cliques) and these mutually interfering cells are the $K-1$ nearest neighbors to the left and to the right. It has been shown (see Appendix) that a valid channel assignment scheme can be found for such systems if all cliques have occupancy less than or equal to the number of channels available for the system, C. In Manhattan model, the co-channel interferers are in both horizontal and vertical directions. In this case, it is proven that the same clique packing conditions are sufficient for a valid channel assignment scheme for the nearest neighbor constraint, i.e. when $K = 2$ [15]. It has also been observed that clique packing can be used as a reasonably good approximation for many network topologies [16, 17].

Thus, since clique packing constraints are easily evaluated (as opposed to computing a channel assignment using graph coloring techniques [18, 19]), the use of clique packing greatly simplifies and accelerates the simulation process and the tenor of the results presented here may be generally applicable to systems with more complicated reuse constraints and topologies.

5 Results and Discussion

5.1 4-cell System

The algorithm was first tested on a small system for which optimal global call admission policies are available via MDP. The system has 4 cells on a ring with nearest neighbor constraint, 4 channels available and is modeled with the new call and handoff service model described in Section 2.

In the first set of experiments, the traffic load of the system is fixed while the relative penalty factor ω for dropping handoff calls is varied. Specifically[2],

$$\lambda = \mu = \gamma = 1.0 \qquad \omega \in [1, 10]$$

The Genetic Algorithm is used to search for local call admission policies with $k = 0$ and 1 that minimize the performance measure given by Equation 1.

In Figure 1, we compare the resulting performances of the best local policies found by GA to the following:

- AIP: admit-if-possible (maximum packing strategy) where a service request is always granted unless it is impossible to assign the channel due to reuse constraints
- OPTIMAL: the optimal global policy performances found by MDP

As expected, weighted blocking for the AIP policy grows linearly with increasing ω. In contrast, the optimal policy weighted blocking rises rapidly and then more slowly. The GA with both $k = 1$ and $k = 0$ provides near-optimal policies for small ω and also outperforms AIP for higher ω. Surprisingly, even completely ignoring neighboring cells ($k = 0$) can result in much better blocking than the AIP strategy.

As ω gets larger, local policy performance degrades relative the optimal. We believe this phenomenon is attributable to a need for information about potential handoffs in distant cells about which the local policy has no knowledge.

The new call blocking and handoff dropping probabilities of the best policies are shown in Figures 2 and 3. The AIP strategy has fixed performances whereas the optimal policy and the policies found by GA tend to increase the new call blocking and favor the handoffs. While the GA found $k = 1$ local policies which do not suggest any regular structure, the $k = 0$ policies found are handoff reservation policies with half of the channels (2 channels) reserved for handoffs for $\omega \geq 5$.

The second set of experiments use fixed loading and penalty factor. In Figure 4, the best performances are

[2] We chose our range for ω based on the folk-rule which states that "handoff dropping is ten times worse than new call blocking."

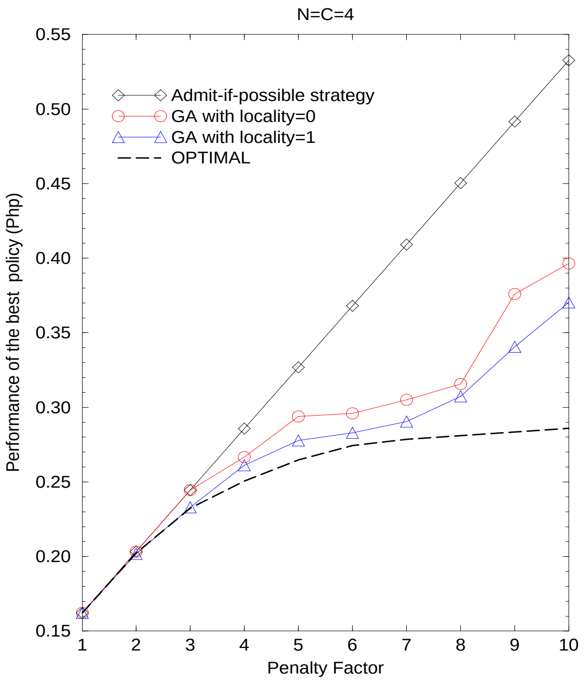

Figure 1: Comparison of Local policy performances to AIP strategy results and the optima. $\lambda = \mu = \gamma = 1.0, \omega, \in [1, 10]$

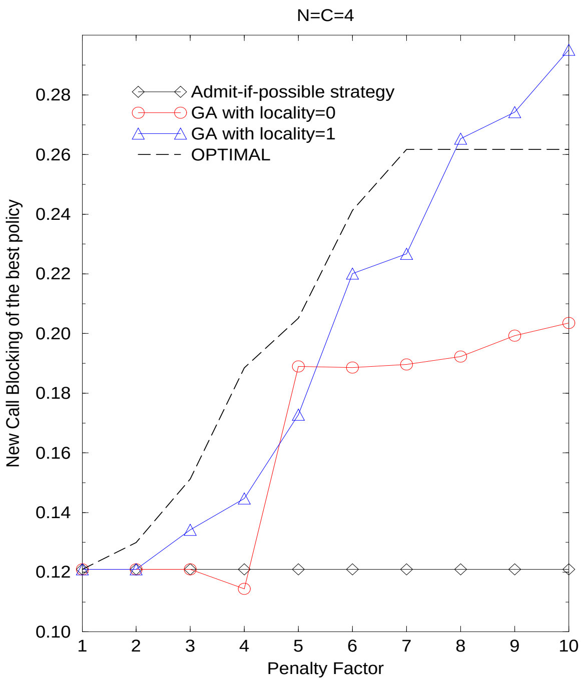

Figure 2: New Call Blocking Performances of the policies for the 4-cell 4-channel system

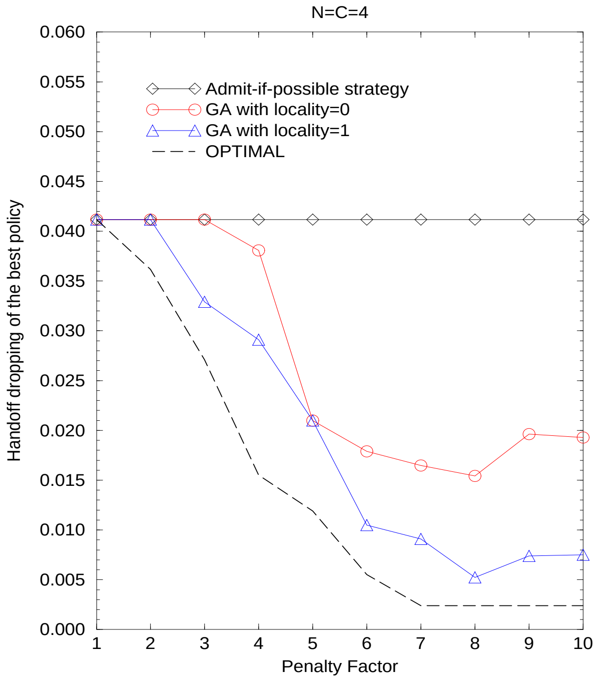

Figure 3: Handoff Dropping Performances of the policies for the 4-cell 4-channel system

found for different handoff rates, γ, using the AIP strategy, GA with $k = 0$ and 1, and compared to the optimal values found by MDP. The exact parameter values are:

$$\lambda = \mu = \rho = 1.0 \qquad \omega = 6$$
$$\gamma \in \{0.1, 0.2, 1.0, 2.0, 3.0, 5.0, 10.0, 20.0\}$$

It is again observed that the GA-derived policies with $k = 0$ and 1 outperform the AIP strategy by a substantial amount and are close in performance to the MDP optimum. The performance using more network information ($k = 1$ vs. $k = 0$) appears to be better, but the gain is not very substantial. As with the variable ω case, better policies (both GA and the optimal) reduce the handoff dropping at the expense of blocking more new calls and begin to degrade as knowledge about potential handoffs in distant cells becomes more important (with increasing γ).

5.2 16-cell System

Next, we consider a larger system with 16 cells and 9 channels again on a ring with nearest neighbor constraint. As explained in Section 3, the number of states in the system is on the order of 10^{16}. Since this exceeds our practical computational limits using MDP, the optimal policy performances are not available for this system.

The search for the best local policies with $k = 0, 1$ for this system is performed using the GA with the following

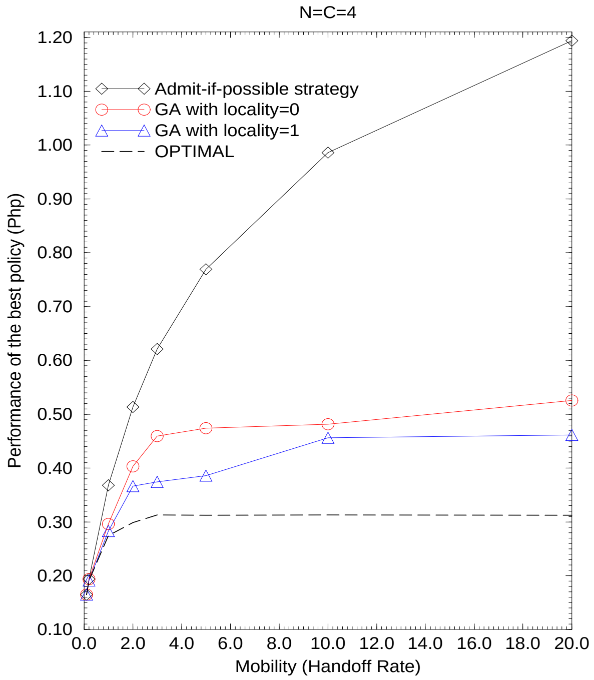

Figure 4: Comparison of the Local policy performances to AIP strategy results and the optima with different mobility rates. $\lambda = \mu = 1, \omega = 6$

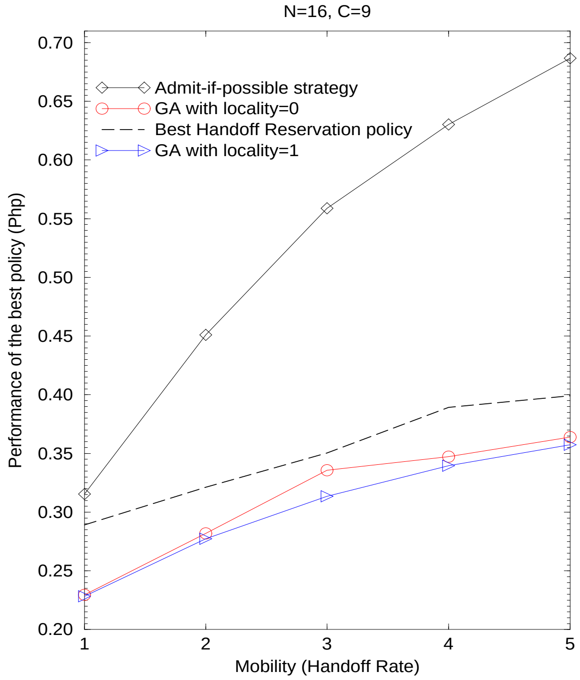

Figure 5: Comparison of the Local policy performances to AIP strategy and the optimum Handoff reservation policies found for the N=16 cell, C=9 channel system with $\lambda = 3, \mu = 1, \omega = 6$

parameters:

$$\lambda = 3.0, \qquad \mu = 1.0, \qquad \omega = 6, \qquad \gamma = 1, 2, \ldots 5$$

The resulting performances of the best policies found by GA are compared to the AIP strategy in Figure 5. In addition, the best handoff reservation policy for this system for each different mobility measure was found using exhaustive search. The handoff reservation policies are defined in the following fashion:

$$D(s) = \begin{cases} 1 & \text{if } s < t \\ 0 & \text{if } s \geq t \end{cases} \tag{3}$$

where s is the cell occupancy (state). $D(s)$ is the decision, admit (1) or reject (0), made by the policy when the cell has s active calls, and t is the threshold beyond which no new call request will be accepted to the cell. The performances of these reservation policies are compared to those for GA-derived local policies in Figure 5.

As with the smaller system, we see that both local policies provide better performance than the AIP strategy. One also expects the local policy search with $k = 1$ to outperform the handoff reservation policies since providing more information cannot degrade optimal policy performance. Specifically, consider that handoff reservation is actually a local policy with $k = 0$; the admission decision is made solely on the basis of cell occupancy and neighbors are ignored.

Note however, that the GA-derived $k = 0$ policy also outperforms the best handoff reservation policy. This result is somewhat surprising since as a general rule, reservation policies often achieve near-optimal performance. To see if the results might be explained by morphological features of the GA-derived polices, we examined the policy structure. The following double threshold of form was observed;

$$D(s) = \begin{cases} 1 & \text{if } s < t_1 \\ 0 & \text{if } t_1 \leq s < t_2 \\ 1 & \text{if } s \geq t_2 \end{cases}$$

as opposed to a single threshold reservation policy defined in Equation 3. Here t_1 and t_2 are the thresholds. For all different mobility rates, the structure of the best local policy with $k = 0$ tends to admit any service request when the cell is not very busy, blocks the new call requests when the cell is moderately busy, and services all requests when the cell is nearly full.

Based on this evidence that double threshold policies may perform better than the reservation policies, we have simulated all double threshold policies to find the best one. The performance comparison between the best handoff reservation policy and the best double threshold policy for a collection of mobility measures is given in Figure 6. We see that imposing the double threshold policy indeed is advantageous relative to a simple handoff reservation policy.

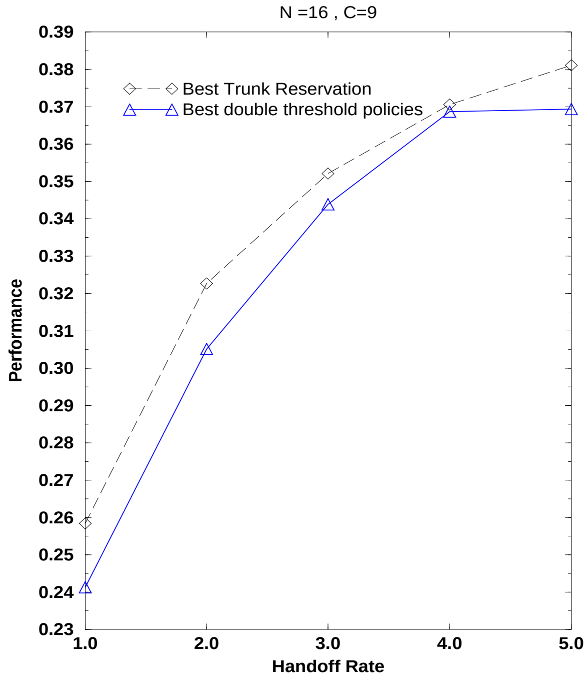

Figure 6: Comparison of the best double threshold policy performances to optimum Handoff reservation policies for the N=16 cell, C=9 channel system with $\lambda = 3, \mu = 1, \omega = 6$

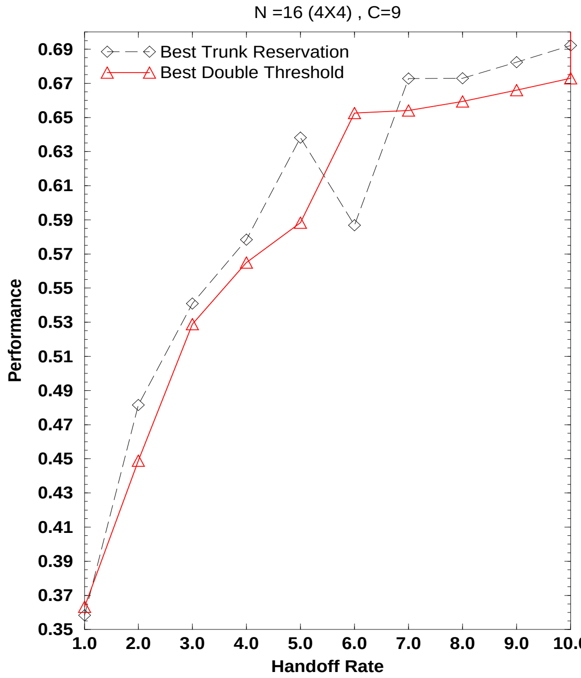

Figure 7: Comparison of the best double threshold policy performances to the optimum Handoff reservation policies found for the N=4X4 cells, C=9 channel system with $\lambda = 3, \mu = 1, \omega = 6$

A moment's thought reveals some method to why this is the case. When the cell is nearly empty and moderately loaded, little is known about the state of neighboring cells and the potential for blocked handoff requests. Thus, the most advisable course is to institute a reservation policy. However, when a cell is nearly full, the reuse constraints require that nearby cells be lightly loaded. This translates into a reduced potential handoff load. Thus, it is advisable to again allow admission to new calls.

More precisely, assume that a cell state (occupancy) n_i is close to the total number of available channels C. Now note that a valid channel assignment for the network is possible only when [13]:

$$n_{i-1} + n_i \leq C \quad \textit{and} \quad n_i + n_{i+1} \leq C$$

So, if $n_i = C - r$, n_{i-1} or n_{i+1} can have at most r active calls. When r is small, cell i is nearly full and the neighboring cells must be nearly *empty*.

The incoming handoff rate to cell i from the neighboring cells is given by:

$$\Gamma = n_{i-1}(0.5\gamma) + n_{i+1}(0.5\gamma) = \gamma(n_{i-1} + n_{i+1})/2$$

$$\Gamma \leq \gamma r$$

Thus, when r is small, Γ (and therefore probability of having a handoff in time Δt, $\Gamma \Delta t$) is small. Now notice that the handoff reservation policy in this case still reserves the channel for an event that is not very probable at the expense of rejecting new calls. The double threshold policies however, suggest allowing access to new call requests when handoffs are improbable and this results in better performance.

5.3 16-cell System in two dimensions

We have also performed an exhaustive search over all double threshold policies and handoff reservation policies and compared the performances for a Manhattan model two dimensional (4X4 cells) cellular network with nearest (horizontal and vertical) neighbor constraint. The results are shown in Figure 7. We see that double threshold policies once again perform better than the handoff reservation policies. The advantage of double threshold policies come into play when the system is nearly full and a handoff event is not very likely due to interference constraints. As explained in Section 5.2, the double threshold policy in this case *does not* reserve a channel for handoff which in effect reduces the new call blocking probability. This effect is shown in Figure 8.

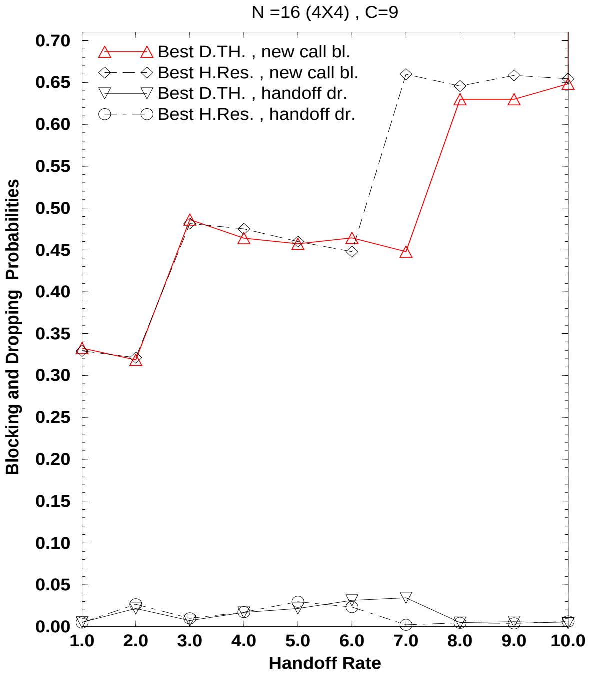

Figure 8: Comparison of the best double threshold policy (Best D.TH.) new call blocking and handoff dropping performances to the best Handoff reservation policies (Best H.Res.) found for the N=4X4 cells, C=9 channel system with $\lambda = 3, \mu = 1, \omega = 6$

6 Conclusion

We have considered the local call admission problem for one dimensional ring-structured and two dimensional Manhattan model cellular networks with two types of service (new call set up and handoff). The system performance measure is defined as a linear combination of new call blocking and handoff dropping probabilities of the network.

To derive policies which minimize this measure, we have used local state-based call admission policies when the optimal policy cannot be found due to computational limitations of MDP method. The search for these local policies is done by a Genetic Algorithm in the one dimensional case.

It was shown that for small systems, local policies performed nearly as well as MDP-derived optimal policies. Since MDP was impractical for larger systems, we used GA to find good admission policies and compared the results to the well-known methods of maximum packing and handoff reservation. The local policies outperformed the maximum packing strategy as well as the best handoff reservation policies. Most striking was the GA ability to identify a novel policy structure which made use of the inherent correlations imposed on neighboring cell occupancy by channel reuse constraints. This result bodes well for the use of GAs as an aid to analytic intuition.

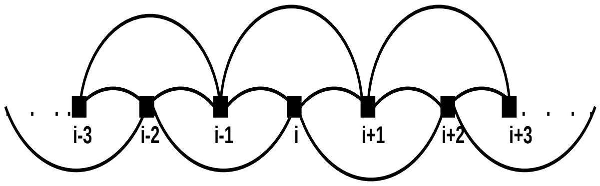

Figure 9: Graph for a one dimensional cellular system with K=3

Finally, we note that from the convergence point of view, the best local policies with $k = 0$ were found much more readily than for $k = 1$.[3] We also note that implementation of $k = 1$ policies would require additional intercell signaling. Coupled to the meager improvement afforded by increasing k from zero to one over the range of handoff blocking weights ω and mobilities γ considered, policies which base their decisions only on single cell occupancy might prove attractive.

A Proof for Clique Packing in one dimension

First we will state the definitions needed for the proof:

Definition 2 *A group of K cells, which are not allowed to use the same channels due to the interference constraints dictated by the system, is called a* clique *of size K.*

The above condition dictates that a cell cannot use the same channels with its $K-1$ nearest neighbors. Note that for a ring structured system, each clique has exactly K cells. The total number of channels available for the system is C.

We number the cliques from left to right, i.e. $clique_i$ represents the clique whose first element is cell i. Let n_j be the number of channels used in cell j, and define the *clique sum* i, Σ_i, as the number of channels used in clique i, i.e.

$$\Sigma_i = \sum_{j \in clique_i} n_j \qquad (4)$$

The system can be represented as a graph with each cell corresponding to a *vertex*. A pair of vertices is connected (share an edge) if they are the members of the same clique. An example is shown in Figure 9.

[3] The policy space size was 2^{30} for $k = 0$ as compared to 2^{3000} for $k = 1$.

Definition 3 *A k-partite graph is one whose vertex set can be partitioned into a minimum of k subsets so that no edge has both ends in any one subset [20].*

To prove if a graph is K partite, one has to construct K disjoint vertex subsets, $V_i(i = 1, ..., K)$, such that $V_i \cap V_j = \emptyset$ for $i \neq j$, and $\cup_{i=1}^{K} V_i = V$, where V is the set of the vertices. Elements of V_i should not be connected to each other.

Note that the K-partition is unique for any graph. Since the sets V_i have to be disjoint and they should span the whole vertex set, the representation of each set is unique. Therefore, if exists, there is only one possible construction of the K-partition.

Now, we make the following claim:

Claim: The graph for a one dimensional ring structured N-cell network with clique size K is K-partite if and only if N is evenly divisible by K.

Proof: Consecutively number the cells from 1 to N and assume that $N \geq K$ (if $N < K$, there are fewer than K vertices and the graph cannot be K-partite). Consider then that any adjacent K cells must lie in different vertex sets. Otherwise, vertices which share an edge will reside in the same vertex set. Thus, the first K cells found *all* the vertex sets if graph is K-partite.

If $N = K$ then we are done and the graph is K-partite. However, if $K < N < 2K$ (N/K not an integer), then notice that cell $K + 1$ shares an edge with the previous $K - 1$ cells *and* with cell 1 since it lies fewer than K cells away. This forces a $(K + 1)^{st}$ vertex set to be defined and the graph cannot be K-partite.

Of course if $N = 2K$ (N/K an integer), then cell $K + 1$ can reside in the first vertex set with cell 1 and the remaining cells $K + i$, $i = 2, 3, ...K$ can reside in vertex sets with cell i.

Now assume that $2K < N < 3K$ (N/K not an integer). The $(2K + 1)^{st}$ cell must reside in the first vertex group containing cells 1 and $K + 1$, otherwise another vertex group must be defined. However, this cell is fewer than K cells away from cell 1 because $N < 3k$ and a $(K + 1)^{st}$ vertex group must be defined so the graph is non K-partite.

Proceeding in this manner for $mK < N < (m + 1)K$ we see that whenever N is not evenly divisible by K, the graph cannot be K-partite. *q.e.d.*

Theorem: For a ring structured one dimensional network whose equivalent graph is K-partite, it is possible to find a channel assignment scheme with C channels if and only if all the clique sums are less than or equal to C.

Proof: Proof for the theorem has two parts:

- ($\Longrightarrow$) :
  This part is trivial. Since violating the interference constraints, i.e. any $\Sigma_i > C$, by definition is not allowed, any legitimate channel allocation scheme has to satisfy the constraints.

- ($\Longleftarrow$) :
  The condition $\Sigma_i \leq C$ is equivalent to card$\{\cup_{j=1}^{K} U_{ji}$ $\} \leq C$ where U_{ji} corresponds to the subset of channels required in cell j of clique i. Therefore, it is a sufficient condition for finding a channel assignment only within the clique.

  We will now assume that the condition, $\Sigma_i \leq C$, is true for all i, and construct a channel assignment scheme as follows. We will assign $U_{11}, \ldots, U_{K1}$ to cells 1,. . .,K. $\Sigma_1 = \sum_{j=1}^{K} \text{card}\{U_{j1}\} \leq C$ is given [4]. The next clique has the same elements except the first cell is replaced by cell $K + 1$, i.e. $clique_1 = 1, 2, \ldots, K$ and $clique_2 = 2, 3, \ldots, K, K + 1$. Cells $2, \ldots, K$ need the same number of channels, i.e.

  $$\Sigma_2 - n_{K+1} = \Sigma_1 - n_1 \tag{5}$$

  Because the equivalent graph is K-partite, cells 1 and $K + 1$ both belong to V_1, i.e. they are not in the same clique. Therefore, channels used in 1 can as well be used in $K + 1$. Note that if the graph were not K-partite, this construction would not be possible. Since we can use the same channels both in cell1 cell $K + 1$, we have $C - \Sigma_1 + n_1$ channels available for cell $K + 1$, and the condition we have to satisfy for a valid channel assignment for $clique_2$ reduces to:

  $$n_{K+1} \leq C - \Sigma_1 + n_1$$

  But, we know by Equation 5 that:

  $$n_{K+1} = \Sigma_2 - \Sigma_1 + n_1 \quad \textit{and} \quad \Sigma_2 \leq C$$

  Therefore,

  $$n_{K+1} \leq C - \Sigma_1 + n_1 \tag{6}$$

  Using the same construction , it can be found that the conditions $n_{K+2} \leq C - \Sigma_2 + n_2, n_{K+3} \leq C - \Sigma_3 + n_3$,...etc. are satisfied for all the cells provided all $\Sigma_i \leq C$. Hence the proof of the second part of the theorem is complete. *q.e.d.*

Corollary: By the Claim, the above Theorem is valid for ring structured cellular networks with N cells, if N is divisible by the clique size K.

References

[1] R.B. Beck and H. Panzer. Strategies for Handover and Dynamic Channel Allocation in Micro-Cellular Mobile Radio Systems. *IEEE Transactions on Communications.*, pages 190–195, April 1989.

[4] card$A$ is the cardinality of set A

[2] M. Zhang and T.S. Yum. Comparison of Channel Assignment Strategies in Cellular Mobile Telephone System. *IEEE Transactions on Vehicular Technology*, pages 211–215, November 1989.

[3] T.J. Kahwa and N.D. Georganas. A hybrid channel assignment scheme in large-scale, cellular structured mobile communication systems. *IEEE Transactions on Communications*, COM-26(4), April 1978.

[4] S. A. Lippman. Applying a new device in the optimization of the exponential queueing systems. *Operations Research*, 23(4), July-August 1975.

[5] J.S. Kaufman. Blocking in a shared resource environment. *IEEE Transactions on Communications*, 29(10), 1981.

[6] K.W. Ross and D. Tsang. The stochastic knapsack problem. *IEEE Transactions on Communications*, 37(7), 1989.

[7] K.W. Ross and D. Tsang. Optimal circuit access policies in an isdn environment: A markov decision approach. *IEEE Transactions on Communications*, 37(9), 1989.

[8] M. Gopalakrishnan and M.K. Sundareshan. Optimal channel allocation policies for access control of circuit-switched traffic in isdn environments. *IEEE Transactions on Communications*, 41(2), 1993.

[9] C. Rose and R. D. Yates. Optimal call admission to single cells of a cellular mobile network. *Winlab Technical Report*, (TR-61), 1993.

[10] A. Yener and C. Rose. Near-optimal call admission policies for cellular networks using genetic algorithms. *Wireless 94, The sixth International Conference on Wireless Communications*, July 1994.

[11] K.S. Meier-Hellstern, E. Alonso, and D. O'Neil. The use of SS7 and GSM to support high density personal communications. In *Proceedings of the International Conference on Communications ICC'92*, 1992.

[12] A. Yener and C. Rose. Local call admission policies for cellular networks using genetic algorithms. *CISS 95, The Twenty-ninth Annual Conference on Information Sciences and Systems*, March 1995.

[13] A. Yener. Finding good call admission policies for cellular mobile networks. *MS Thesis, Rutgers University*, 1994.

[14] D. A. Goldberg. *Genetic Algorithms in search, optimization and machine learning*. Addison-Wesley, 1989.

[15] M. Andersin and M. Frodigh. Traffic adaptive channel assignment in city environments. In *Proceedings of the Fourth WINLAB Workshop on Third Generation Wireless Networks, p.280*, 1994.

[16] P. Raymond. Performance analysis of cellular networks. *IEEE Transactions on Communications*, COM-39:1787–1793, 1991.

[17] D.L. Pallant and P.G. Taylor. Approximation of performance measures in cellular networks with dynamic channel allocation. *Teletraffic Research Centre Report, Univ. of Adelaide*, 1992.

[18] L.A. McGeoch D.S. Johnson, C.R. Araon and C. Schevon. Optimiazation by Simulated Annealing: An Experimental Evaluation; Part II, Graph Coloring and Number Partitioning. *Operations Research*, 39, 1991.

[19] K.N. Sivarajan, R.J. McEliece, and J.W. Ketchum. Channel Assignment in Cellular Radio. In *Proceedings of the VTC*, pages 846–850. IEEE, 1989.

[20] J. A. Bondy and U. S. R. Murty. *Graph Theory with Applications*. North Holland.