Optimal Admission Control Using Handover Prediction in Mobile Cellular Networks*

Vicent Pla, José Manuel Giménez-Guzmán†, Jorge Martínez and Vicente Casares-Giner
Department of Communications
Universidad Politécnica de Valencia (UPV)
ETSIT Camí de Vera s/n, 46022 Valencia, Spain.
Telephone: +34 963879733, Fax: +34 963877309
E-mail: (vpla,jmartinez,vcasares)@dcom.upv.es
jogiguz@doctor.upv.es

Abstract

In this paper we study the impact of incorporating handover prediction information into the call admission control process in mobile cellular networks. The comparison is done between the performance of optimal policies obtained with and without the predictive information. The prediction agent classifies mobile users in the neighborhood of a cell into two classes, those that will probably be handed over into the cell and those that probably will not. We consider the classification error by modeling the false-positive and non-detection probabilities. Two different approaches to compute the optimal admission policy were studied: *dynamic programming* and *reinforcement learning*. Preliminary results show significant performance gains when the predictive information is used in the admission process.

1 Introduction

Future mobile communication systems are expected to support broadband multimedia services with diverse Quality of Service (QoS) requirements. The cellular architecture is used in wireless networks to utilize the radio spectrum efficiently. Since mobile users may change cells a number of times during the lifetime of their connections, availability of wireless network resources at the connection setup time does not necessarily guarantee that wireless network resources are available throughout the lifetime of a connection. Thus users may experience a performance degradation due to mobile handovers. This problem is magnified by the current trend to reduce the cell size to accommodate more mobile users in a given area as handover events will occur at a much higher rate [1].

In addition to packet-level QoS issues (like packet delay, jitter and error probability) connection-level issues (related to connection establishment and management) need to be addressed in mobile wireless networks [2]. A Call Admission Control (CAC) policy is required to address this problem.

CAC in single service cellular systems has been thoroughly studied, see for instance the seminal work by Hong and Rappaport [3] or more recent papers like [4–6] and references therein. While most of these papers provide intuitive reservation schemes for the CAC a more insightful approach is adopted in [7] and [8], where the CAC is regarded as an optimization problem. Admission control in the presence of mobility and multiple services is not that well studied although some works in this direction can be found in the literature [9–12].

On the other hand, most of the proposed CAC policies take the admission decision using only state information local to the cell such as the number of active connections per service type. However, mobile cellular networks permit to have some anticipated knowledge about forthcoming requests, and more importantly, this predictive information concerns the most sensible requests, namely, the handover attempts. Following that observation, several mobility prediction (MP) schemes and associated CAC policies have appeared (see [2, 13–16] and references therein). A common feature of these studies is the proposal of heuristic CAC policies to exploit the specific information provided by each MP scheme.

*This work has been supported by the *Spanish Ministry of Science and Technology* under projects TIC2001-0956-C04-04 and TIC2003-08272.

†This author was supported by the Universidad Politécnica de Valencia under *Programa de Incentivo a la Investigación*.

In this paper we study the CAC problem using predictive information from an optimization perspective in both single service and multiservice scenarios. Our goal is to obtain the optimal policy for a given amount of information provided by the MP scheme. We consider that such approach has not been sufficiently explored. In [17] the authors determine a near-optimal policy by means of a genetic algorithm that takes into account not only the cell state but also the state of the neighboring cells, in a single service scenario. However, results in [17] show that the performance gain when using these additional information is rather insignificant. We reached the same conclusion using a different optimization method. These disappointing results suggest that the prediction of possible forthcoming handovers obtained from the occupancy state of the neighboring cells is not sufficiently specific. Here we try to go a step further and we evaluate the performance gain that can be obtained when the CAC process is provided with more specific information. In our study the total population of active terminals in the surroundings of the cell is divided into two classes: those that will handover into the cell with high probability and those that will not. Our model of the prediction agent (PA) does not provide information about the time instant when the handover will occur. We postpone the study of this scenario for a future work. Obviously, the more information is provided by the prediction agent (PA) the better the performance of the CAC policy will be. Unfortunately, the complexity of the PA and the optimization process increases as more information is provided.

The rest of the paper is structured as follows. In Section 2 we describe the model of the system and that of PA. The two optimization approaches and some numerical results are presented in Section 3. Finally, a summary of the paper a some concluding remarks are given in Section 4.

2 Model Description

We consider a cell with a total of C resource units, being the physical meaning of a unit of resources dependent on the specific technological implementation of the radio interface. A total of N different classes of services are offered by the system. For each type of service new and handover call arrivals are distinguished so that there are N types of services and $2N$ types of arrivals.

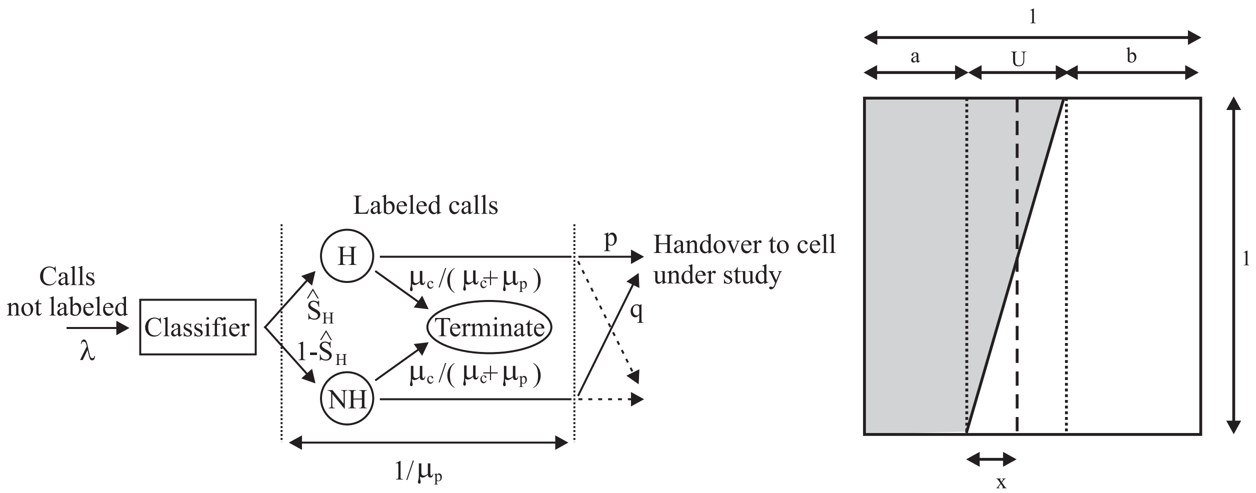
For the sake of mathematical tractability we make the common assumptions of Poisson arrival processes and exponentially distributed random variables for cell residence time and call duration. The arrival rate for new (handover) calls of service i is λ_i^n (λ_i^h) and a request of service i consumes b_i resource units, $b_i \in \mathbb{N}$. The call duration of service i is exponentially distributed with rate μ_i^c. The cell residence time of a service i customer is exponentially distributed with rate μ_i^r. Hence, the resource holding time in a cell for service i is exponentially distributed with rate $\mu_i = \mu_i^c + \mu_i^r$.

2.1 Prediction Agent

Given that the focus of our study was not the design of the PA we used a model of it instead. The PA informs the CAC about the number of active terminals in the neighborhood that are forecasted to produce a handover into the cell. The amount of time elapsed since an active mobile terminal (MT) is deemed as "probably producing a handover" until the handover actually occurs is not predicted by the PA and we model it by an exponential random variable. In general, the classification into "probably producing a handover" (H) or the opposite (NH) is not completely accurate and therefore we incorporate the probabilities of non-detection and false-positive, which is shown in Fig. 1.

An active MT entering the cell neighborhood is labeled by the PA as H or NH according to some of its characteristics (position, trajectory, velocity, historic profile, ...) and/or some other information (road map, hour of the day, ...). After an exponentially distributed time, the actual destiny of the MT becomes definitive and either a handover into the cell occurs or not (for instance because the connection ends or the MT moves to another cell). The CAC system is aware at any time of the number of MTs labeled as H.

Our model of the PA is characterized by three parameters: the average sojourn time of the MT in the predicted stage μ_p^{-1}, the probability p of producing a handover if labeled as H and the probability q of producing a handover if labeled as NH. Note that in general $q \neq 1-p$. The values of p and q relate to each other through the specific model of the PA (see Fig. 1(b)). In the figure there is a square (with a surface equal to one) representing the population of MTs that is going to be classified by the PA. The shaded area represents the fraction of MTs that will ultimately produce a handover into the cell while the white area represents the rest of MTs. The



(a) Basic operation of the prediction scheme (b) Proposed model of the PA

Figure 1: Model of the PA.

classifier sets a threshold, which is represented by a vertical dashed line, to discriminate between those MTs that will likely produce a handover and those that will not. MTs falling on the left side of the threshold are labeled as H and those on the right side as NH. There exists an uncertainty zone, which is represented by the slope of the line separating the shaded and white areas. Parameter x represents the relative position of the classifier threshold. This uncertainty produces classification errors: the white area on the left of the threshold and the shaded area on the right of the threshold.

Let us introduce the following notation referring to the areas in Fig. 1(b): S_H denotes the shaded surface and represents the proportion of MTs that will produce a handover; $\hat{S}_H$ denotes the surface on left side of the threshold and is the proportion of MTs labeled as H; $\hat{S}^e_H$ denotes the white surface on left side of the threshold and represents the proportion of MTs labeled as H that will not produce a handover; $\hat{S}^e_{NH}$ denotes the shaded surface on the right side of the threshold and represents the proportion of MTs labeled as NH that will produce a handover. From Fig. 1(b) it follows that

$$1 - p = \frac{\hat{S}^e_H}{\hat{S}_H} = \frac{x^2}{2U(a+x)} \; ; \qquad q = \frac{\hat{S}^e_{NH}}{1 - \hat{S}_H} = \frac{(U-x)^2}{2U(1-a-x)}$$

Parameters a and b can be expressed in terms of the proportion of handovers to the target cell S_H and the degree of uncertainty in the prediction U,

$$a = S_H - U/2 \; ; \qquad b = 1 - S_H - U/2$$

and then

$$1 - p = \frac{\hat{S}^e_H}{\hat{S}_H} = \frac{x^2}{U(2S_H - U + x)} \; ; \qquad q = \frac{\hat{S}^e_{NH}}{1 - \hat{S}_H} = \frac{(U-x)^2}{U(1 - 2S_H + U - x)}$$

3 Optimization of the Admission Policy

The information provided by the PA and the state of the cell (number of active connections) is used to find the optimal admission policy and its performance. The performance metric that we use is a weighted sum of the loss rate for each arrival type. The arrival type is defined by two components: whether it is a new connection request or a handover request for an ongoing connection, and the service type of the connection.

The whole system is modeled as a *Markov decision process* (MDP) [18]. Two different optimization approaches have been used to find the optimal policy. The first approach is based on *dynamic programming* (DP) [18], specifically we used a policy improvement method. This approach is applied to a single service scenario. The second is an automatic learning approach based on the theory of *reinforcement learning* [19], more specifically we used the average reward reinforcement learning algorithm proposed in [20]. This approach is applied to a multiservice scenario. DP gives an exact solution and allows to evaluate the theoretical limits of incorporating movement prediction in the CAC problem, whereas RL tackles more efficiently the curse of dimensionality and offers the important advantage of being a model-free method, i.e. transition probabilities

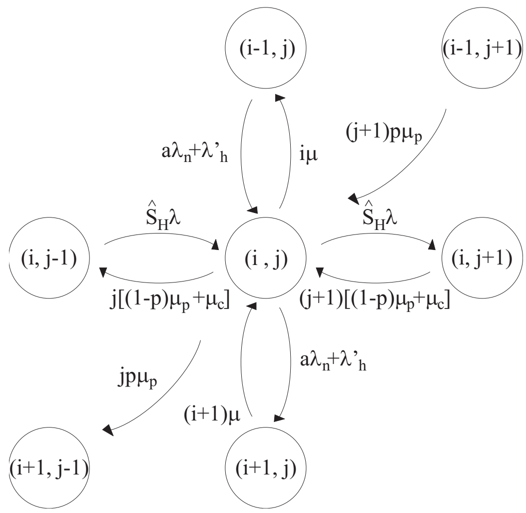


Figure 2: Transition rates.

and average costs are not needed by the method. As a consequence, in the RL approach, neither the numerical values of the PA parameters (p, q, μ_p) nor the arrival rates and holding times need to be known beforehand. Moreover, the learning algorithm can adapt to variations of those parameters.

3.1 Single Service

In this section we describe the optimization approach based on DP. Since there is only one service thought this section we simplify notation by omitting the subscript referring to the service type, i.e. $\lambda_n = \lambda_1^n$, $\lambda_h = \lambda_1^h$, $\mu_c = \mu_1^c$, $\mu_r = \mu_1^r$, $\mu = \mu_1$. We assume without loss of generality that $b_1 = 1$.

Let us represent the system state by (i, j) where i is the number of active calls in the cell and j is the number of MTs labeled as H. The set of possible states of the system is

$$S := \Big\{\boldsymbol{x} = (i, j) : 0 \le i \le C;\ 0 \le j \le C_p\Big\}$$

where C_p represents the maximum number of MT that can be labeled as H at a given time. We use large value for C_p so that it has no practical impact in our results. At each state $(i, j), i < C$, the set of possible actions is defined by $A := \{a : a = 0, 1\}$, being $a = 0$ the action that rejects an incoming new call and $a = 1$ the action that accepts an incoming new call. Handover calls have priority over new call and they are accepted as long as there are available resources ($i < C$). At state (C, j) only the action $a = 0$ is possible.

Figure 2 shows the transition rates from and to state (i, j). Note that some of the transition rates depend on the decision $a = 0, 1$. In the figure we introduced λ'_h which is the average arrival rate of handovers that have not been predicted, and it is given by

$$\lambda'_h = (1 - \hat{S}_H)\frac{\mu_p}{\mu_p + \mu_c} q\lambda$$

where λ is the input rate to the PA.

The model described is a continuous-time Markov chain, we will convert it to a *Discrete Time Markov Chain* (DTMC) by applying uniformization (see [21, Section 4.7]). It can be shown that $\Gamma = C_p(\mu_p + \mu_c) + C(\mu_r + \mu_c) + \lambda + \lambda_n$ is a uniform upper-bound for the outgoing rate of all the states. If $r_{\boldsymbol{xy}}(a)$ denotes the transition rate from state $\boldsymbol{x}$ to state $\boldsymbol{y}$ when action a is taken at state $\boldsymbol{x}$, then the transition probabilities of the resulting DTMC are given by

$$p_{\boldsymbol{xy}}(a) = \frac{r_{\boldsymbol{xy}}(a)}{\Gamma} \quad \text{if } \boldsymbol{y} \neq \boldsymbol{x} \qquad \text{and} \qquad p_{\boldsymbol{xx}}(a) = 1 - \sum_{\boldsymbol{y} \in S} p_{\boldsymbol{xy}}(a)$$

We define the incurred cost at state $\boldsymbol{x}$ when action a is taken by

$$C(\boldsymbol{x}, a) = \begin{cases} (1-a)\lambda_n, & i < C,\ a = 0, 1 \\ \lambda_n + \beta(\lambda'_h + jp\mu_p), & i = C,\ a = 0 \end{cases}$$

The weighting factor β (typically) accounts for the fact that blocking a handover request is less desirable than blocking of a new call request. Costs are defined so that the expected average cost $L(\pi)$ equals a weighted sum of the average loss rate of new calls $L_n(\pi)$ and the average loss rate of handover attempts $L_h(\pi)$, i.e.

$$L(\pi) = L_n(\pi) + \beta L_h(\pi) = \lim_{n\to\infty} E\left[\frac{1}{n+1}\sum_{t=0}^{n} C\big(\boldsymbol{x}(t), \pi(\boldsymbol{x}(t))\big)\right]$$

where $\boldsymbol{x}(t)$ is the state visited at time t when policy π is deployed. Loss rates are given by

$$L_n(\pi) = \sum_{\boldsymbol{x}:\ \pi(\boldsymbol{x})=0} \lambda_n p(\boldsymbol{x})\ ; \qquad L_h(\pi) = \sum_{\substack{\boldsymbol{x}=(C,j)\\ 0\le j\le C_p}} \big(S_H q\lambda + jp\mu_p\big)p(\boldsymbol{x})$$

where $p(\boldsymbol{x})$ is the stationary probability of state $\boldsymbol{x}$. Thus, the optimization problem pursues to find the policy (π^*) that minimizes $L(\pi)$. Since state $(0,0)$ can be reached from any other state regardless of the policy deployed, by virtue of the *Corollary 6.20* and the subsequent remark, both in [18], we know that an optimal stationary policy exists.

Let $l_{\boldsymbol{x}}(\pi)$ denote the relative cost of state $\boldsymbol{x}$ under policy π, we can write

$$l_{\boldsymbol{x}}(\pi) = C(\boldsymbol{x}, \pi(\boldsymbol{x})) - L(\pi) + \sum_{\boldsymbol{y}} p_{\boldsymbol{xy}}(\pi(\boldsymbol{x}))l_{\boldsymbol{y}}(\pi) \qquad \forall \boldsymbol{x} \tag{1}$$

from which we can obtain the average cost and the relative costs $l_{\boldsymbol{x}}(\pi)$ up to an undetermined constant. Thus we arbitrarily set $l_{(0,0)}(\pi) = 0$ and then solve the linear system of equations (1) to obtain $L(\pi)$ and $l_{\boldsymbol{x}}(\pi),\ \forall\boldsymbol{x}$. Having obtained the average and relative costs under policy π an improved policy π' can be calculated as

$$\pi'(\boldsymbol{x}) = \arg\min_{a=0,1}\Big\{C(\boldsymbol{x}, a) - L(\pi) + \sum_{\boldsymbol{y}} p_{\boldsymbol{xy}}(a)l_{\boldsymbol{y}}(\pi)\Big\}$$

so that the following relation holds $L(\pi') \le L(\pi)$. Moreover, if the equality holds then $\pi' = \pi = \pi^*$, where π^* denotes the optimal policy, i.e. $L(\pi^*) \le L(\pi)\ \forall\pi$.

If we repeat iteratively the solution of system (1) and the policy improvement until we obtain a policy which does not change after improvement. This process is called *Policy Iteration* [22, Section 8.6] and it leads to the average optimal policy in a finite — and typically small — number of iterations.

3.1.1 Numerical Results

In this section we compare the expected average cost $L(\pi)$ for the optimal policies with and without predictive information. When no predictive information is used the optimization is carried without considering the second components of the system state, i.e. the number of MT labeled as H, and the optimal policy results to be of the *guard channel* type [7].

In our numerical results we used the following settings, unless otherwise indicated: $C = 10$, $C_p = 60$, $N_h = \mu_r/\mu_c = 2$, $\mu_p^{-1}/\mu_r^{-1} = 0.5$, $\beta = 20$, $x = U/2$, $S_H = 0.4$, $\lambda_n = 1$. The value of λ is chosen so that the system is in statistical equilibrium, i.e. the rate at which handoff calls enter a cell is equal to the rate at which handoff calls exit the cell, $\lambda = 0.989(N_h + \mu_p^{-1}/\mu_r^{-1})\lambda_n/S_H$.

The curves in Figs. 3 through 6 represent the quotient between the performance of the optimal policy when no prediction is deployed and the optimal policy deploying prediction. As expected, using prediction induces a gain in all cases and that gain decreases as prediction uncertainty (U) increases. In Fig. 3 we varied the average number of handovers per call. In Fig. 4 we varied the weighting factor β which quantifies the priority of handover requests over new calls: the higher the value of β the lower the blocking probability of handover calls compared to the blocking probability of new calls. It is observed that higher values of β lead to higher performance gains. The position of the decision threshold within the uncertainty zone is evaluated in Fig. 5, the curves indicate that a threshold in the middle of the the uncertainty zone is the best choice. Finally, Fig. 6 shows the effect of the elapse time since an MT is classified as H until it is handed over into the target cell or it moves to another cell. Both, short and long prediction periods, have a negative effect on the performance gain.

In all the cases that we examined the optimal policy had a *dynamic guard channel* structure, in which the number of reserved channels increases with the number of MTs labeled as H. More formally, let $p(i,j)$ be the probability of accepting a new call when the system is at state (i,j), then

$$p(i,j) = \begin{cases} 1, & \text{if } i \le i_{th}(j) \\ 0, & \text{if } i > i_{th}(j) \end{cases} \qquad \text{and} \quad i_{th}(j) \le i_{th}(j') \quad \text{if} \quad j > j'.$$

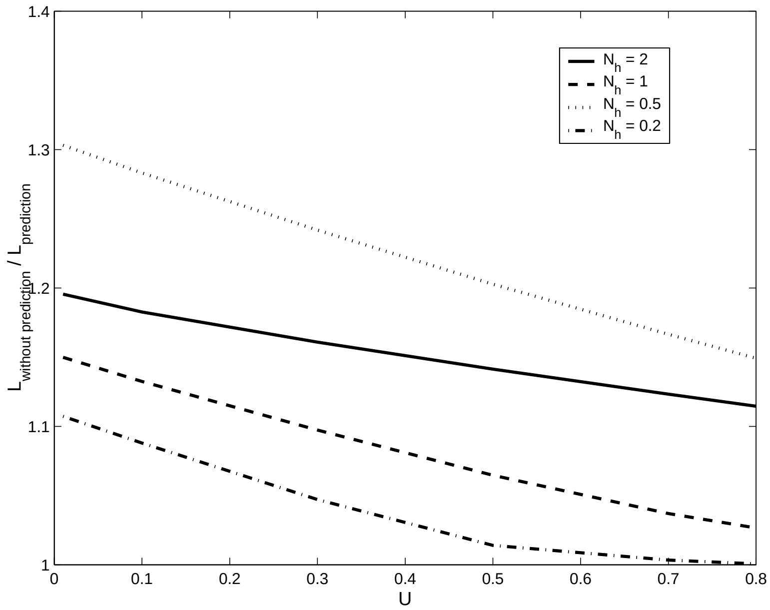


Figure 3: Performance comparison in a single service scenario. Impact of mobility, N_h.

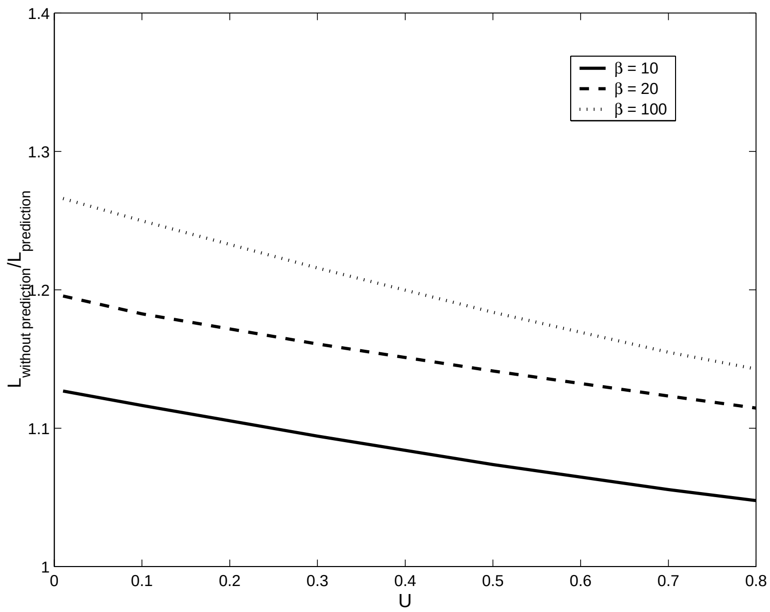


Figure 4: Performance comparison in a single service scenario. Impact of weighting factor, β.

where $i_{th}(j)$ is the threshold for a given j.

3.2 Multiservice

In this section we describe the optimization approach based on *reinforcement learning* (RL). We define a single performance parameter $L(\pi)$ the average expected cost rate under policy π, which value can be determined by

$$L(\pi) = \sum_{i=1}^{N} (\omega_i^n P_i^n \lambda_i^n + \omega_i^h P_i^h \lambda_i^h)$$

where $\omega_i^{n,h}$ represent the relative weights associated to the blocking of a new or handover requests of service i and $P_i^{n,h}$ represent the loss probabilities of new or handover requests of service i. In general, $\omega_i^n < \omega_i^h$ to account for the fact that the blocking of a handover request is less desirable than the blocking of a new call request.

We studied two scenarios in which the information available to the CAC system is of increasing value. In all the scenarios we considered that the number of arrival types is $2N$. In the first scenario the CAC system is provided with the state information of the cell neighborhood, e.g. a vector with N elements, each of them

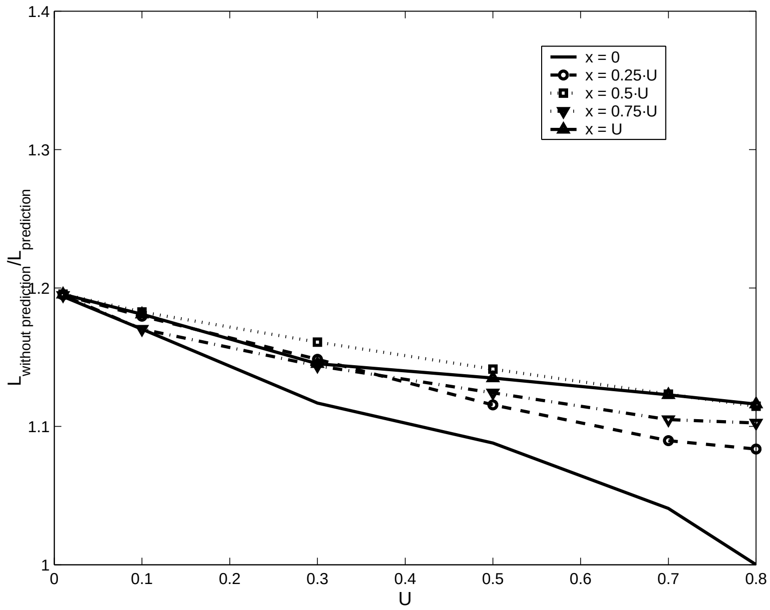


Figure 5: Performance comparison in a single service scenario. Impact of decision threshold, x/U.

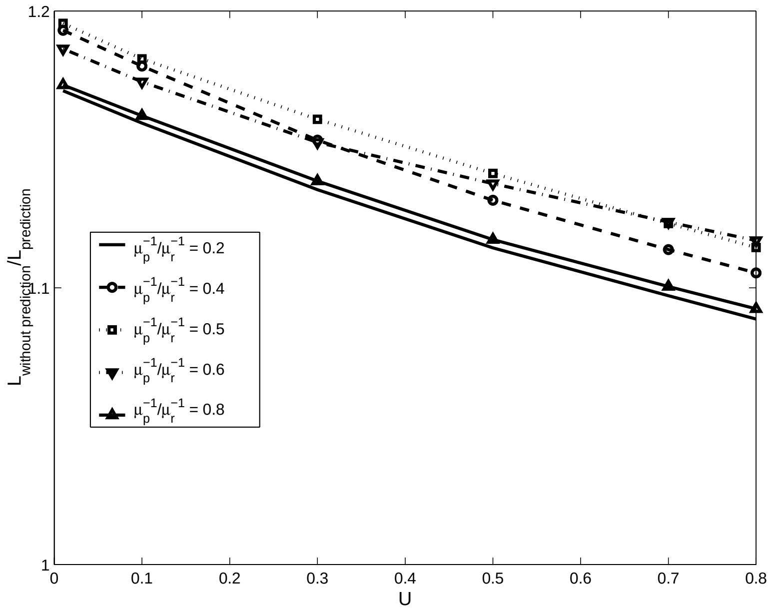


Figure 6: Performance comparison in a single service scenario. Impact of anticipation time, μ_p^{-1}/μ_r^{-1}.

being the number of ongoing calls of a given service in the surrounding of the cell. In a second scenario the CAC system is also provided with the state information of the cell neighborhood, but the elements of the state vector represent the number of ongoing calls of each arrival type labeled as H by the PA.

We formulate the optimization problem as an infinite-horizon finite-state semi-Markov decision process (SMDP) under the average cost criterion, which is more appropriate for the problem under study than other discounted cost approaches [23]. It is evident that we search for policies that minimize L. The decision epochs correspond to the time instants in which an arrival occurs. Given that no actions are taken at call departures, then only the arrival events are relevant to the optimization process. Additionally, we select one of the $2N$ arrival types as the highest priority one, being its requests always admitted while free resources are available, and therefore no decisions are taken for them. The generic definition of the system state vector is $(x_1, \ldots, x_N, x_{N+1}, \ldots, x_{2N}, k)$, where x_i is the number of ongoing calls of service i, $1 \leq i \leq N$, in the cell under study, x_{i+N} is the number of ongoing calls of service i in cell neighborhood and k, $1 \leq k \leq (2N-1)$, represents the arrival type. At each decision epoch the system has to select an action from the set of possible actions $A := \{0 = \text{reject}, 1 = \text{admit}\}$. Therefore, the state space can now be defined as

$$S := \left\{ \boldsymbol{x} = (x_1, \dots, x_N, x_{N+1}, \dots, x_{2N}, k) : x_i \in \mathbb{N}; \sum_{i=1}^{N} x_i b_i \le C; \sum_{i=N+1}^{2N} x_i b_i \le C_p \right\}.$$

A deterministic stationary policy is a mapping from states to actions, $\pi : S \to A$, that defines the next action based only on the current state $\boldsymbol{x}$. Starting from any initial state and following policy π, the system evolves as a semi-Markov process with transition probabilities given by $P_{\boldsymbol{xy}}(a)$, which represent the probability of moving from state $\boldsymbol{x}$ to state $\boldsymbol{y}$ under action $a = \pi(\boldsymbol{x})$.

The cost structure is defined as follows. At any decision epoch, the cost incurred by accepting any arrival type is zero and by rejecting a new (handover) request of service i is ω_i^n (ω_i^h). With this framework, further accrual of cost occurs when the system has to reject requests of the highest priority arrival type between two decision epochs. If we call $w(\boldsymbol{x}, a, \boldsymbol{y})$ the finite cost for executing action a in state $\boldsymbol{x}$ and resulting in state $\boldsymbol{y}$ at the next decision epoch, and we suppose that using a policy π the system evolves through states $\boldsymbol{x}_0, \boldsymbol{x}_1, \dots, \boldsymbol{x}_t$ in interval $[0, t]$ then the total cost accumulated in the interval is $w^\pi(x_0, t) = \sum_{m=0}^{t-1} w\big(x_m, \pi(x_m), x_{m+1}\big)$.

If the environment is stochastic then $w^\pi(x_0, t)$ is a random variable. Under the average cost criterion we seek to minimize the average expected cost rate over time t, as $t \to \infty$. When the system starts at state $\boldsymbol{x}$ and follows policy π, this is denoted by $g^\pi(\boldsymbol{x})$ and is defined as: $g^\pi(\boldsymbol{x}) = \lim_{t \to \infty} \frac{1}{t} E\left[w^\pi(\boldsymbol{x}, t)\right]$.

In our system, it is not difficult to see that for every deterministic stationary policy the embedded Markov chain has a unichain transition probability matrix, and therefore the average expected cost rate does not vary with the initial state [22]. We call it the "cost" of the policy π, denote it by g^π and consider the problem of finding the policy π^* that minimizes g^π, which we name the optimum policy. It is a simple matter to show that $g^\pi = L(\pi)$. Even though the cost of a policy g^π is independent of the initial state, the total expected reward at time t may not be so. We denote by $h(\boldsymbol{x})$ the *bias* of state $\boldsymbol{x}$ which can be interpreted as the expected long-term advantage in total cost for starting in state $\boldsymbol{x}$ in addition to $g^\pi t$, the expected total cost in time t on the average. The Bellman optimality recurrence equations for an SMDP under the average cost criterion has the form

$$h^*(\boldsymbol{x}) = \min_{a \in A_x} \left\{ w(\boldsymbol{x}, a) - g^* \tau(\boldsymbol{x}, a) + \sum_{\boldsymbol{x} \in S} P_{\boldsymbol{xy}}(a) h^*(\boldsymbol{y}) \right\}$$

where $h^*(\boldsymbol{x})$ is an optimal state dependent relative value function (bias) and $\tau(\boldsymbol{x}, a)$ is the average sojourn time in state $\boldsymbol{x}$ under action a. The greedy policy π^* defined by selecting actions that minimize the right-hand side of the above equation is gain-optimal [20].

If the parameters of the model can be derived, then the solution to the Bellman equations can be obtained through dynamic or linear programming techniques. In multiservice scenarios, where the number of states can be large, the derivation of the model parameters can be complex and make the problem intractable (course of dimensionality). We propose an alternative approach based on a reinforcement learning algorithm named Semi-Markov Average Reward Technique (SMART) [20].

The Bellman equations can be rewritten as

$$h^*(\boldsymbol{x}, a) = \min_{a \in A_x} \left\{ w(\boldsymbol{x}, a) - g^* \tau(\boldsymbol{x}, a) + \sum_{\boldsymbol{x} \in S} P_{\boldsymbol{xy}}(a) \min_{a' \in A_{\boldsymbol{y}}} h^*(\boldsymbol{y}, a') \right\}$$

where $h^*(\boldsymbol{x}, a)$ is the average expected relative value of taking the optimal action a in state $\boldsymbol{x}$ and then continuing indefinitely by choosing actions optimally. Then we have that the optimal policy is $\pi^*(\boldsymbol{x}) = \arg\min_{a \in A_x} h^*(\boldsymbol{x}, a)$.

The SMART algorithm estimates $h^*(\boldsymbol{x}, a)$ by simulation using a temporal difference method (TD(0)). If at the $(m-1)^{th}$ decision epoch the system is in state $\boldsymbol{x}$ and action a is taken, and the system is found in state $\boldsymbol{y}$ at the m^{th} decision epoch then we update the relative state-action values as follows:

$$h_{new}(\boldsymbol{x}, a) = (1 - \alpha_m) h_{old}(\boldsymbol{x}, a) + \alpha_m \left\{ w_m(\boldsymbol{x}, a, \boldsymbol{y}) - g_m \tau_m(\boldsymbol{x}, a, \boldsymbol{y}) + \min_{a' \in A_{\boldsymbol{y}}} h_{old}(\boldsymbol{y}, a') \right\}$$

where $w_m(\boldsymbol{x}, a, \boldsymbol{y})$ is the actual cumulative cost incurred between the two successive decision epochs, $\tau_m(\boldsymbol{x}, a, \boldsymbol{y})$ is the actual sojourn time between the decision epochs, α_m is the learning rate parameter at the m^{th} decision

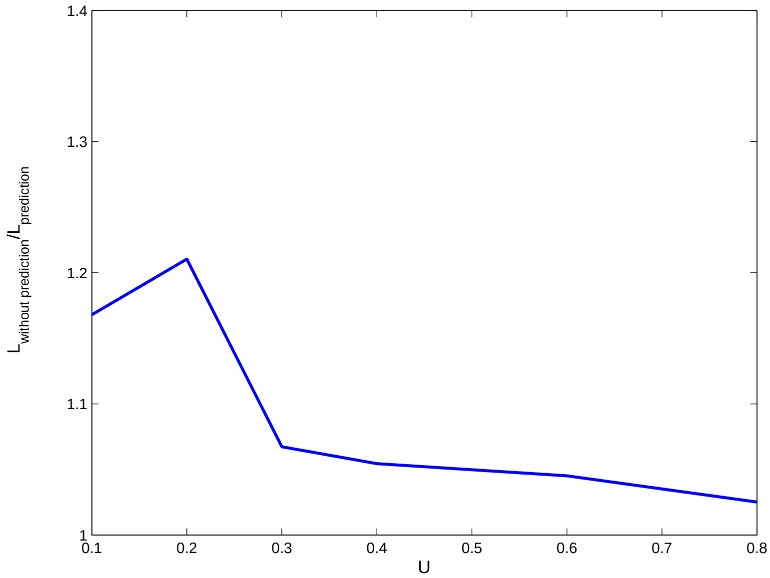


Figure 7: Performance comparison in a multiservice scenario.

epoch, and g_m is the average cost rate estimated as:

$$g_m = \frac{\sum_{k=1}^{m} w_k\left(\boldsymbol{x}_{(k)}, a_{(k)}, \boldsymbol{y}_{(k)}\right)}{\sum_{k=1}^{m} \tau_k\left(\boldsymbol{x}_{(k)}, a_{(k)}, \boldsymbol{y}_{(k)}\right)}$$

3.2.1 Numerical results

We evaluated the performance gain in a scenario with the following characteristics: $C = 10$ and $C_p = 60$ resource units, $N = 2$ services, $b_1 = 1$ and $b_2 = 2$ resource units, $N_H = 1$, $\lambda = 1.54$, $\lambda_1^n = 0.8\lambda$, $\lambda_2^n = 0.2\lambda$, $\lambda_1^h = 2.352\lambda$, $\lambda_2^h = 0.588\lambda$, $\mu_1 = 1$, $\mu_2 = 3$, $w_1^n = 1$, $w_2^n = 20$, $w_1^h = 9.8$, $w_2^h = 196$, $\hat{S}_H = 0.4$ and $x = U/2$. In regards to the reinforcement learning algorithm, we use a constant learning rate $\alpha = 0.01$ but the exploration rate p_m is decayed to zero by using the following rule $p_m = p_0/(1+u)$, where $u = m^2/(\gamma + m)$. We used $\gamma = 2.5 \cdot 10^8$ to obtain $p_m = 1 \cdot 10^{-3} p_0$ when $m = 5 \cdot 10^5$. We start with an exploration rate $p_0 = 0.1$.

Figure 7 shows the variation of the quotient between the performance of the optimal policy when no prediction is deployed and the optimal policy deploying prediction for different values of the uncertainty U. For each value of U we run 10 simulations with different seeds and we display the average of the values obtained for $L_{prediction}$. As expected, using prediction induces a gain in all cases and that gain decreases as the prediction uncertainty (U) increases. The figure shows a peak at $U = 0.2$ but we consider that it is not relevant as it could be due to the fact that reinforcement learning is only able to find suboptimal solutions.

4 Conclusion

In this paper we analyze the performance gain that can be obtained when handover prediction information is considered in order to optimize the admission control policy in a mobile cellular network. Predictive information is provided by a prediction agent that labels the active mobile terminals in the neighborhood of the cell which will probably produce a handover into the cell. The policy optimization has been performed in a Markov or semi-Markov decision process framework and two optimization methods have been applied: policy iteration and a model free reinforcement learning methods. Our numerical results show that typical performance gains are around 10% although improvement ratios up to 30% have also been observed in some specific scenarios. In future work we will consider a more sophisticated model of the prediction agent including, for instance, a more precise estimation of the time instant in which a handover will occur.

References

[1] W. Y. Lee, "Smaller cells for greater performance," *IEEE Communications Magazine*, pp. 19–23, Nov. 1991.

[2] S. Choi and K. G. Shin, "Predictive and adaptive bandwidth reservation for handoffs in QoS-sensitive cellular networks," in *Proceedings of ACM SIGCOMM'98*, Sept. 1998, pp. 155–166.

[3] D. Hong and S. S. Rappaport, "Traffic model and performance analysis for cellular mobile radio telephone systems with prioritized and nonprioritized handoff procedures," *IEEE Transactions on Vehicular Technology*, vol. VT-35, no. 3, pp. 77–92, Aug. 1986, see also: CEAS Technical Report No. 773, June 1, 1999, College of Engineering and Applied Sciences, State University of New York, Stony Brook, NY 11794, USA.

[4] F. Barceló, "Performance analysis of handoff resource allocation strategies through the state-dependent rejection scheme," *IEEE Transactions on Wireless*, vol. 3, no. 3, pp. 900–909, May 2004.

[5] D. J. Daley and L. D. Servi, "Loss probabilities of hand-in traffic under various protocols: II. model comparisons," *Performance Evaluation*, vol. 55, no. 3-4, pp. 231–249, Feb. 2004.

[6] V. Pla and V. Casares-Giner, "Analysis of priority channel assignment schemes in mobile cellular communication systems: a spectral theory approach," *Performance Evaluation*, 2004, in press.

[7] R. Ramjee, R. Nagarajan, and D. Towsley, "On optimal call admission control in cellular networks," *Wireless Networks Journal (WINET)*, vol. 3, no. 1, pp. 29–41, 1997.

[8] N. Bartolini, "Handoff and optimal channel assignment in wireless networks," *Mobile Networks and Applications (MONET)*, vol. 6, no. 6, pp. 511–524, 2001.

[9] N. Bartolini and I. Chlamtac, "Call admission control in wireless multimedia networks," in *Proceedings of IEEE PIMRC*, 2002.

[10] H. Heredia-Ureta, F. A. Cruz-Pérez, and L. Ortigoza-Guerrero, "Capacity optimization in multiservice mobile wireless networks with multiple fractional channel reservation," *IEEE Transactions on Vehicular Technology*, vol. 52, no. 6, pp. 1519 – 1539, Nov. 2003.

[11] V. Pla and V. Casares-Giner, "Optimal admission control policies in multiservice cellular networks," in *Proceedings of the International Network Optimization Conference (INOC)*, 2003, pp. 466–471.

[12] D. García, J. Martínez, and V. Pla, "Comparative evaluation of admission control policies in cellular multiservice networks," in *Proceedings of the 16th International Conference on Wireless Communications*, July 2004.

[13] D. Levine, I. Akyildiz, and M. Naghshineh, "A resource estimation and call admission algorithm for wireless multimedia networks using the shadow cluster concept," *IEEE/ACM Transactions on Networking*, vol. 5, no. 1, pp. 1–12, Feb. 1997.

[14] J. Hou and Y. Fang, "Mobility-based call admission control schemes for wireless mobile networks," *Wireless Communications and Mobile Computing*, vol. 1, no. 3, pp. 269–282, 2001.

[15] F. Yu and V. Leung, "Mobility-based predictive call admission control and bandwidth reservation in wireless cellular networks," *Computer Networks*, vol. 38, no. 5, pp. 577–589, 2002.

[16] W.-S. Soh and H. S. Kim, "Dynamic bandwidth reservation in cellular networks using road topology based mobility prediction," in *Proceedings of IEEE INFOCOM*, 2004.

[17] C. Yener, A. Rose, "Genetic algorithms applied to cellular call admission: local policies," *IEEE Transactions on Vehicular Technology*, vol. 46, no. 1, pp. 72–79, 1997.

[18] S. M. Ross, *Applied probability models with optimization applications*. Holden-Day, 1970.

[19] R. Sutton and A. G. Barto, *Reinforcement Learning*. Cambridge, Massachusetts: The MIT press, 1998.

[20] T. K. Das, A. Gosavi, S. Mahadevan, and N. Marchalleck, "Solving semi-markov decision problems using average reward reinforcement learning," *Management Science*, vol. 45, no. 4, pp. 560–574, 1999.

[21] R. W. Wolff, *Stochastic Modeling and the Theory of Queues*. Englewood Cliffs, NJ: Prentice Hall, 1989.

[22] M. L. Puterman, *Markov Decision Processes : Discrete Stochastic Dynamic Programming*. John Wiley & Sons, 1994.

[23] S. Mahadevan, "Average reward reinforcement learning: Foundations, algorithms, and empirical results," *Machine Learning, Special Issue on Reinforcement Learning (edited by Leslie Kaebling)*, vol. 22, no. 1–3, pp. 159–196, Jan./Feb./March 1996.