

Performance Bounds for Mobile Cellular Networks with Handover Prediction

Jose Manuel Gimenez-Guzman, Jorge Martinez-Bauset, and Vicent Pla

Departamento de Comunicaciones, Universidad Politecnica de Valencia , UPV
ETSIT Camino de Vera s/n, 46022, Valencia, Spain
jogiguz@doctor.upv.es, {jmartinez, vpla}@dcom.upv.es

Abstract. We determine the gain that can be achieved by incorporating movement prediction information in the session admission control process in mobile cellular networks. The gain is obtained by evaluating the performance of optimal policies achieved with and without the predictive information, while taking into account possible prediction errors. We evaluate the impact of predicting only incoming handovers, only outgoing or both types together. The prediction agent is able to determine the handover instants both stochastically and deterministically. Two different approaches to compute the optimal admission policy were studied: dynamic programming and reinforcement learning. Numerical results show significant performance gains when the predictive information is used in the admission process, and that higher gains are obtained when deterministic handover instants can be determined.

1 Introduction

Session Admission Control (SAC) is a key aspect in the design and operation of mobile cellular networks that provide QoS guarantees. Terminal mobility makes it very difficult to guarantee that the resources available at the time of session setup will be available in the cells visited during the session lifetime, unless a SAC policy is exerted. The design of the SAC system must take into account not only packet level issues (like delay, jitter or losses) but also session level issues (like loss probabilities of both session setup and handover requests). This paper explores the second type of issues from a novel optimization approach that exploits the availability of movement prediction information. To the best of our knowledge, applying optimization techniques to this type of problem has not been sufficiently explored. The results provided define theoretical limits for the gains that can be expected if handover prediction is used, which could not be established by deploying heuristic SAC approaches.

In systems that do not have predictive information available, both heuristic and optimization approaches have been proposed to improve the performance of the SAC at the session level. A optimization approach without using predictive information has been studied in [1,2,3,4]. In systems that have predictive information available, most of the proposed approaches to improve performance are heuristic, see for example [5,6] and references therein.

Our work has been motivated in part by the study in [5]. Briefly, the authors propose a sophisticated movement prediction system and a SAC scheme that taking advantage of movement prediction information is able to improve system performance. One of the novelties of the proposal is that the SAC scheme takes into consideration not only incoming handovers to a cell but also the outgoing ones. The authors justify it by arguing that considering only the incoming ones would lead to reserve more resources than required, given that during the time elapsed since the incoming handover is predicted and resources are reserved until it effectively occurs, outgoing handovers might have provided additional free resources, making the reservation unnecessary.

This paper can be considered an extension of the work presented in [7], incorporating new contributions. One of them is the comparative performance evaluation of incorporating different types of predictive information to the SAC optimization process, like only incoming, only outgoing and both types of handovers together. In [7] only the incoming handover prediction was studied. Another contribution is the evaluation of the impact that predicting deterministically the future handover instants have on the system performance. In [7] only stochastic prediction was modeled.

In a previous study [7] we considered a scenario with several service types and no qualitative differences were found between single and multiservice cases. On the other hand, the higher complexity of multiservice scenarios could hide the insight into the performance implications of using handover prediction information, which is the focus of this paper.

The rest of the paper is structured as follows. In Section 2 we describe the models of the system and the two prediction agents deployed. The two optimization approaches are presented in Section 3. A numerical evaluation comparing the performance obtained when using different types of information and when handovers instants are deterministically or stochastically predicted is provided in Section 4. Finally, a summary of the paper and some concluding remarks are given in Section 5.

2 Model Description

We consider a single cell system and its neighborhood, where the cell has 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. Only one service is offered but new and handover session arrivals are distinguished, making a total of two arrival types.

For mathematical tractability we make the common assumptions. New and handover sessions arrive according to a Poisson process with rates λ_n and λ_h respectively. The duration of a session and the cell residence time are exponentially distributed with rates μ_s and μ_r respectively, hence the resource holding time in a cell is also exponentially distributed with rate $\mu = \mu_s + \mu_r$. Without loss of generality, we will assume that each session consumes one unit of resource and that only one session is active per MT.

We used a model of the prediction agent, given that the focus of our study was not the design of it.

2.1 Prediction Agent for Incoming Handovers

An active MT entering the cell neighborhood is labeled by the prediction agent for incoming handovers (IPA) as “probably producing a handover” (H) or the opposite (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 session ends or the MT moves to another cell) as shown in Fig. 1(a). The SAC system is aware of the number of MTs labeled as H at any time.

The model of the classifier is shown in Fig. 1(b) where the square (with a surface equal to one) represents the population of active MTs to be classified. The shaded area represents the fraction of MTs (S_H) that will ultimately move into the cell, while the white area represents the rest of active MTs. Notice that part of the MTs that will move into the cell can finish their active sessions before doing so. The classifier sets a threshold (represented by a vertical dashed line) to discriminate between those MTs that will likely produce a handover and those that will not. The fraction of MTs falling on the left side of the threshold (\hat{S}_H) are labeled as H and those on the right side as NH. There exists an uncertainty zone, of width U , which produces classification errors: the white area on the left of the threshold (\hat{S}_H^e) and the shaded area on the right of the threshold (\hat{S}_{NH}^e). The parameter x represents the relative position of the classifier threshold within the uncertainty zone. Although for simplicity we use a linear model for the uncertainty zone it would be rather straightforward to consider a different model.

As shown in Fig. 1(a), the model of the IPA 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 $1 - p$ and q model the

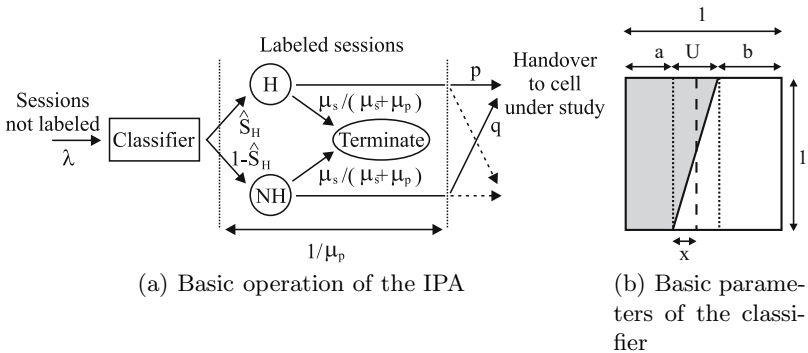


Fig. 1. IPA and classifier models

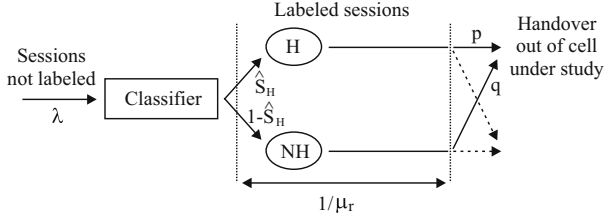


Fig. 2. Basic operation of the OPA

false-positive and non-detection probabilities and in general $q \neq 1 - p$. It can be shown that

$$1 - p = \frac{\hat{S}_H^e}{\hat{S}_H} = \frac{x^2}{(U(2S_H - U + 2x))}; \quad q = \frac{\hat{S}_{NH}^e}{(1 - \hat{S}_H)} = \frac{(U - x)^2}{(U(2 - 2S_H + U - 2x))} \quad (1)$$

2.2 Prediction Agent for Outgoing Handovers

The model of the prediction agent for outgoing handovers (OPA) is shown in Fig. 2. The OPA labels active sessions in the cell as H if they will produce a handover or as NH otherwise. The classification is performed for both handover sessions that enter the cell and new sessions that initiate in the cell, and are carried out by a classifier which model is the same as the one used in the IPA. The time elapsed since the session is labeled until the actual destiny of the MT becomes definitive is the cell residence time that, as defined, is exponentially distributed with rate μ_r . The fraction of sessions that effectively execute an outgoing handover is given by $S_H = \mu_r / (\mu_s + \mu_r)$. The OPA model is characterized by only two parameters $1 - p$ and q , which meaning is the same as in the IPA model. Note that $1 - p$ and q can be related to the classifier parameters by the expressions in (1).

3 Optimizing the SAC Policy

We formulate the optimization problem as an infinite-horizon finite-state Markov decision process under the average cost criterion, which is more appropriate for the problem under study than other discounted cost approaches. When the system starts at state \mathbf{x} and follows policy π then the average expected cost rate over time t , as $t \rightarrow \infty$, is denoted by $\gamma^\pi(\mathbf{x})$ and defined as: $\gamma^\pi(\mathbf{x}) = \lim_{t \rightarrow \infty} \frac{1}{t} E[w^\pi(\mathbf{x}, t)]$, where $w^\pi(\mathbf{x}, t)$ is a random variable that expresses the total cost incurred in the interval $[0, t]$. For the systems we are considering, 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 [8]. We call it the ‘‘cost’’ of the policy π , denote it by γ^π and consider the problem of finding the policy π^* that minimizes γ^π , which we name the optimal policy.

In our model the cost structure is chosen so that the average expected cost represents a weighted sum of the loss rates, i.e. $\gamma^\pi = \omega_n P_n \lambda_n + \omega_h P_h \lambda_h$, where ω_n (ω_h) is the cost incurred when the loss of a new (handover) request occurs and P_n (P_h) is the loss probability of new (handover) requests. In general, $\omega_n < \omega_h$ since the loss of a handover request is less desirable than the loss of a new session setup request.

Two different optimization approaches have been used to find the optimal SAC policy: a dynamic programming (DP) approach and an automatic learning approach based on the theory of Reinforcement Learning (RL) [9]. DP gives an exact solution and allows to evaluate the theoretical limits of incorporating handover prediction in the SAC system, whereas RL tackles more efficiently the curse of dimensionality. In both approaches handover sessions have priority over new sessions and they are accepted as long as resources are available.

3.1 Dynamic Programming

We apply DP to the scenario that only considers the incoming handovers, in which case the system state space is $S := \{\mathbf{x} = (i, j) : 0 \leq i \leq C; 0 \leq j \leq C_p\}$, where i is the number of active sessions in the cell, j is the number of MTs labeled as H in the cell neighborhood and C_p is the maximum number of MT that can be labeled as H at a given time. We use a 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 session and $a = 1$ the action that accepts an incoming new session. The system can be described as a continuous-time Markov chain which state transition diagram is shown in Fig. 3, where $\lambda'_h = q\lambda(1 - \hat{S}_H)\mu_p / (\mu_p + \mu_s)$ denotes the average arrival rate of unpredicted handovers. It is converted to a *discrete time Markov chain* (DTMC) by applying uniformization. It can be shown that $\Gamma = C_p(\mu_p + \mu_s) + C(\mu_r + \mu_s) + \lambda + \lambda_n$ is an uniform upper-bound

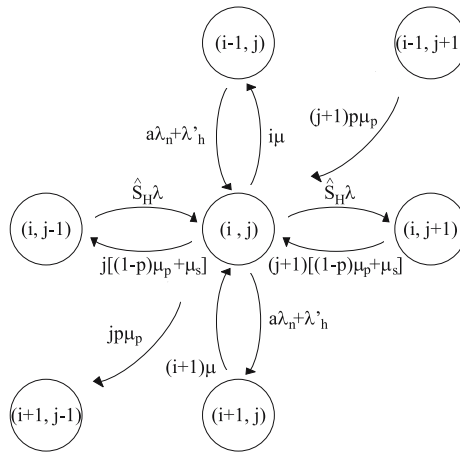


Fig. 3. State transition diagram

for the outgoing rate of all the states, being λ the input rate to the classifier. If $r_{\mathbf{x}\mathbf{y}}(a)$ denotes the transition rate from state \mathbf{x} to state \mathbf{y} when action a is taken at state \mathbf{x} , then the transition probabilities of the resulting DTMC are given by $p_{\mathbf{x}\mathbf{y}}(a) = r_{\mathbf{x}\mathbf{y}}(a)/\Gamma$ ($\mathbf{y} \neq \mathbf{x}$) and $p_{\mathbf{x}\mathbf{x}}(a) = 1 - \sum_{\mathbf{y} \in S} p_{\mathbf{x}\mathbf{y}}(a)$. We define the incurred cost rate at state \mathbf{x} when action a is selected by $c(\mathbf{x}, a)$, which can take any of the following values: 0 ($i < C$, $a = 1$), $\omega_n \lambda_n$ ($i < C$, $a = 0$) or $\omega_n \lambda_n + \omega_h (\lambda'_h + jp\mu_p)$ ($i = C$, $a = 0$).

If we denote by $h(\mathbf{x})$ the relative cost rate of state \mathbf{x} under policy π , then we can write

$$h(\mathbf{x}) = c(\mathbf{x}, \pi(\mathbf{x})) - \gamma^\pi + \sum_{\mathbf{y}} p_{\mathbf{x}\mathbf{y}}(\pi(\mathbf{x})) h(\mathbf{y}) \quad \forall \mathbf{x} \quad (2)$$

from which we can obtain the average cost and the relative costs $h(\mathbf{x})$ up to an undetermined constant. Thus we arbitrarily set $h(0, 0) = 0$ and then solve the linear system of equations (2) to obtain γ^π and $h(\mathbf{x})$, $\forall \mathbf{x}$. Having obtained the average and relative costs under policy π an improved policy π' can be calculated as

$$\pi'(\mathbf{x}) = \arg \min_{a=0,1} \left\{ c(\mathbf{x}, a) - \gamma^\pi + \sum_{\mathbf{y}} p_{\mathbf{x}\mathbf{y}}(a) h(\mathbf{y}) \right\}$$

so that the following relation holds $\gamma^{\pi'} \leq \gamma^\pi$. Moreover, if the equality holds then $\pi' = \pi = \pi^*$, where π^* denotes the optimal policy, i.e. $\gamma^{\pi^*} \leq \gamma^\pi \forall \pi$.

We repeat iteratively the solution of system (2) and the policy improvement until we obtain a policy which does not change after improvement. This process is called *Policy Iteration* [8, Section 8.6] and it leads to the average optimal policy in a finite — and typically small — number of iterations.

3.2 Reinforcement Learning

We formulate the optimization problem as an infinite-horizon finite-state semi-Markov decision process (SMDP) under the average cost criterion. The decision epochs correspond only to the time instants at which new session arrivals occur, given no decisions are taken for handover arrivals. Only arrival events are relevant to the optimization process because no actions are taken at session departures. The state space for the scenario that only considers the incoming handovers is defined as $S := \{\mathbf{x} = (x_0, x_{in}) : x_0 \leq C; x_{in} \leq C_p\}$, where x_0 and x_{in} represent, respectively, the number of resource units occupied by sessions in the cell and by sessions in the neighborhood which are labeled as H. The state space for the scenario that only considers the outgoing handovers is defined as $S := \{\mathbf{x} = (x_0, x_{out}) : x_0, x_{out} \leq C\}$, where x_{out} represents the number of resource units occupied by sessions in the cell labeled as H. The state space for the scenario that considers both the incoming and outgoing handovers is defined as $S := \{\mathbf{x} = (x_0, x_{in}, x_{out}) : x_0, x_{out} \leq C; x_{in} \leq C_p\}$. At each decision epoch the system has to select an action from the set $A := \{0, 1\}$.

The cost structure is defined as follows. At any decision epoch, the cost incurred by accepting a new session request is zero and by rejecting it is ω_n . Further accrual of cost occurs when the system has to reject handover requests between two decision epochs, incurring a cost of ω_h per rejection.

The Bellman optimality recurrence equations for a SMDP under the average cost criterion can be written as

$$h^*(\mathbf{x}, a) = \min_{a \in A_x} \{w(\mathbf{x}, a) - \gamma^* \tau(\mathbf{x}, a) + \sum_{\mathbf{y} \in S} p_{\mathbf{x}\mathbf{y}}(a) \min_{a' \in A_y} h^*(\mathbf{y}, a')\}$$

where $h^*(\mathbf{x}, a)$ is the average expected relative cost of taking the optimal action a in state \mathbf{x} and then continuing indefinitely by choosing actions optimally, $w(\mathbf{x}, a)$ is the average cost of taking action a in state \mathbf{x} , $\tau(\mathbf{x}, a)$ is the average sojourn time in state \mathbf{x} under action a and $p_{\mathbf{x}\mathbf{y}}(a)$ is the probability of moving from state \mathbf{x} to state \mathbf{y} under action $a = \pi(\mathbf{x})$. The greedy policy π^* defined by selecting actions that minimize the right-hand side of the above equation is gain-optimal [10].

In systems where the number of states can be large, RL tackles more efficiently the curse of dimensionality and offers the important advantage of being a model-free method, i.e. transition probabilities and average costs are not needed in advance. We deploy the SMART algorithm [10], which estimates $h^*(\mathbf{x}, a)$ by simulation using a temporal difference method (TD(0)). If at the $(m-1)^{th}$ decision epoch the system is in state \mathbf{x} , action a is taken and the system is found in state \mathbf{y} at the m^{th} decision epoch then we update the relative state-action values as follows: $h_{new}(\mathbf{x}, a) = (1 - \alpha_m)h_{old}(\mathbf{x}, a) + \alpha_m\{w_m(\mathbf{x}, a, \mathbf{y}) - \gamma_m \tau_m(\mathbf{x}, a, \mathbf{y}) + \min_{a' \in A_y} h_{old}(\mathbf{y}, a')\}$, where $w_m(\mathbf{x}, a, \mathbf{y})$ is the actual cumulative cost incurred between the two successive decision epochs, $\tau_m(\mathbf{x}, a, \mathbf{y})$ is the actual sojourn time between the decision epochs, α_m is the learning rate parameter at the m^{th} decision epoch and γ_m is the average cost rate estimated as: $\gamma_m = \sum_{k=1}^m w_k(\mathbf{x}(k), a(k), \mathbf{y}(k)) / \sum_{k=1}^m \tau_k(\mathbf{x}(k), a(k), \mathbf{y}(k))$.

4 Numerical Evaluation

We evaluated the performance gain when introducing prediction by the ratio $\gamma_{wp}^\pi / \gamma_p^\pi$, where γ_p^π (γ_{wp}^π) is the average expected cost rate of a policy that is optimal in a system with (without) prediction. We assume a circular-shaped cell of radio r and a holed-disk-shaped neighborhood with inner (outer) radio $1.0r$ ($1.5r$).

The values of the parameters that define the scenario are: $C = 10$ and $C_p = 60$, $N_h = \mu_r / \mu_s = 1$, $\mu_r / \mu_p = 0.5$, $\lambda_n = 2$, $\mu = \mu_s + \mu_r = 1$, $S_H = 0.4$, $x = U/2$, $w_n = 1$, and $w_h = 20$. The value of the input rate to the PA λ is chosen so that the system is in statistical equilibrium, i.e. the rate at which handover sessions enter the cell is equal to the rate at which handover sessions exit the cell. It can be easily shown that for our scenario $\lambda = (1 - P_n)(1 - P_{ft})\lambda_n(N_h + \mu_r / \mu_p)(1/S_H)$, where $P_{ft} = P_h / (P_h + \mu_s / \mu_r)$ is the probability of forced termination. Note that in our numerical experiments the values of the arrival rates are chosen to achieve realistic operating values for P_n ($\approx 10^{-2}$) and P_{ft} ($\approx 10^{-3}$). For such values, we make the approximation $\lambda \approx 0.989\lambda_n(N_h + \mu_r / \mu_p)(1/S_H)$.

For the RL simulations, the ratio of arrival rates of new sessions to the cell neighborhood (ng) and to the cell (nc) is made equal to the ratio of their surfaces,

$\lambda_{ng} = 1.25\lambda_{nc}$. The ratio of handover arrival rates to the cell neighborhood from the outside of the system (ho) and from the cell (hc) is made equal to ratio of their perimeters, $\lambda_{ho} = 1.5\lambda_{hc}$. Using the flow equilibrium property, we can write $\lambda_{hc} = (1 - P_n)(1 - P_{ft})(\mu_r/\mu_s)\lambda_{nc} \approx 0.989(\mu_r/\mu_s)\lambda_{nc}$. With regard to the RL algorithm, we use a constant learning rate $\alpha_m = 0.01$ but the exploration rate p_m is decayed to zero by using the following rule $p_m = p_0/(1 + u)$, where $u = m^2/(\varphi + m)$. We used $\varphi = 1.0 \cdot 10^{11}$ and $p_0 = 0.1$. The exploration of the state space is a common RL technique used to accept non-improving solutions in order to avoid being trapped at local minima.

Figure 4 shows the gain when introducing prediction for different values of the uncertainty U . When using RL, for each value of U we run 10 simulations with different seeds and display the averages. As observed, using incoming handover prediction induces a gain and that gain decreases as the prediction uncertainty (U) increases.

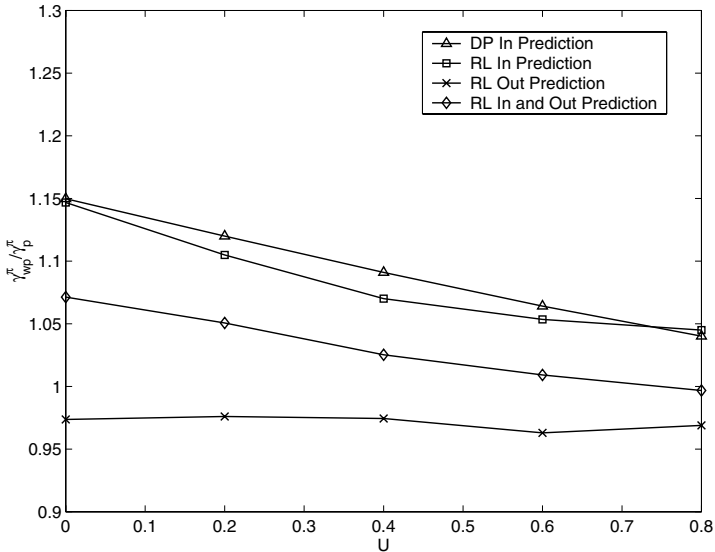


Fig. 4. Performance gain when using stochastic handover prediction

From Fig. 4 it is clear that the knowledge of the number of resources that will become available is not relevant for the determination of optimum SAC policies, being even independent of the degree of uncertainty. It can also be observed that the optimization algorithm finds slightly worse solutions when using information related to the outgoing handovers ($\gamma_{wp}^\pi/\gamma_p^\pi < 1$). This is probably due to the difficulty that the algorithm has to find good solutions in a bigger space state. This observation seems to be corroborated when comparing the results obtained using only the incoming handover information and using both the incoming and outgoing handover information together. As shown, the solutions found in the second case are slightly worse than the ones found in the first one.

4.1 Deterministic Prediction

The prediction agents described in Sections 2.1 and 2.2 predict the time instants at which handovers will occur only stochastically. In this section we evaluate the impact on performance that more precise knowledge of the future handover time instants have. Intuitively, it seems obvious that handovers taking place in a near future would be more relevant for the SAC process than those occurring in an undetermined far future. More precisely, in this section both the IPA and OPA operate as before but they label the sessions T time units before handovers take place, i.e. the component x_{in} (x_{out}) of the different state spaces represent the number of incoming (outgoing) handovers that will take place in less than T time units. A similar approach is used in [5], where authors predict the incoming and outgoing handovers that will take place in a time window of fixed size.

For the performance evaluation, the same scenarios, parameters and methodology described before in this same Section where used, except that we set the uncertainty to a constant value $U = 0.2$, which we consider it might be a practical value. Figure 5 shows the variation of the gain for different values of T . As observed, there exists an optimum value for T , which is close to the mean time between call arrivals (λ^{-1}), although it will probably depend on other system parameters as well. As T goes beyond its optimum value, the gain decreases, probably because the temporal information becomes less significant for the SAC decision process. As expected, when $T \rightarrow \infty$ the gain is identical to the one in the stochastic prediction case because the labeling of sessions occur at the same time instants, i.e. when handover sessions enter the cell or new sessions are initiated in the cell. When T is lower than its optimum value the gain also decreases, probably because the system has not enough time to react. When $T = 0$ the gain is null because there is no prediction at all.

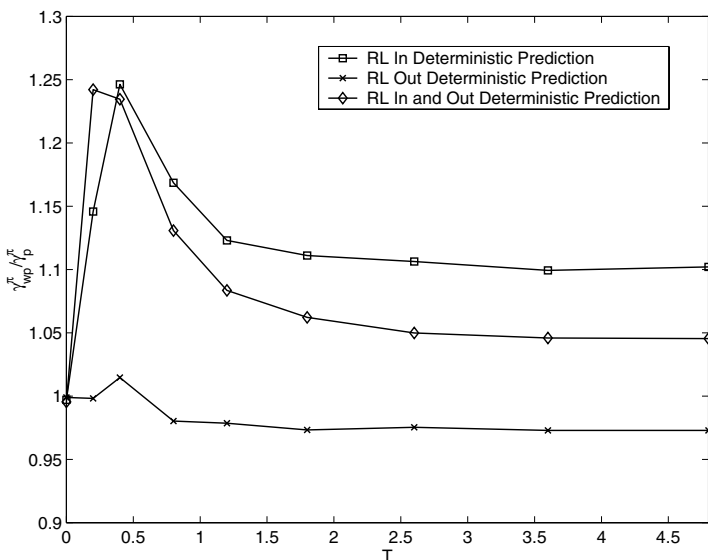


Fig. 5. Performance gain when using deterministic handover prediction

Figure 5 shows that the information provided by the OPA is again not relevant for the optimization process. For values of T close to its optimum the gain is similar when using incoming handover prediction or incoming and outgoing prediction together, and it is significantly higher than when stochastic time prediction is used.

In an earlier version of the IPA we were providing the optimization process with state information of the neighboring cells and obtained that the gain was not significant, possibly because the information was not sufficiently specific. The authors in [11] reached the same conclusion but using a genetic algorithm to find near-optimal policies. In the version described in this paper, we are providing the optimization process with state information of a sufficiently close neighborhood, obtaining significant gains. For the design of the OPA we were faced with the same dilemma but in this case we decided not to use more specific information. Defining a holed-disk-shaped neighborhood with outer (inner) radius r ($< r$) for the outgoing handovers and an exponentially distributed sojourn time in it, would had open the possibility of having terminals that could go in and out of this area, making the cell residence time not exponential. This would had made the models with the IPA and with the OPA not comparable. Besides, providing the optimization process with more specific information of the outgoing handovers does not help to improve the performance either, as observed in Fig. 5.

Finally it is worth noting that the main challenge in the design of efficient bandwidth reservation techniques for mobile cellular networks is to balance two conflicting requirements: reserving enough resources to achieve a low forced termination probability and keeping the resource utilization high by not blocking too many new setup requests. Figure 6, which shows the utilization gain for different values of U , justifies the efficiency of our optimization approach. It

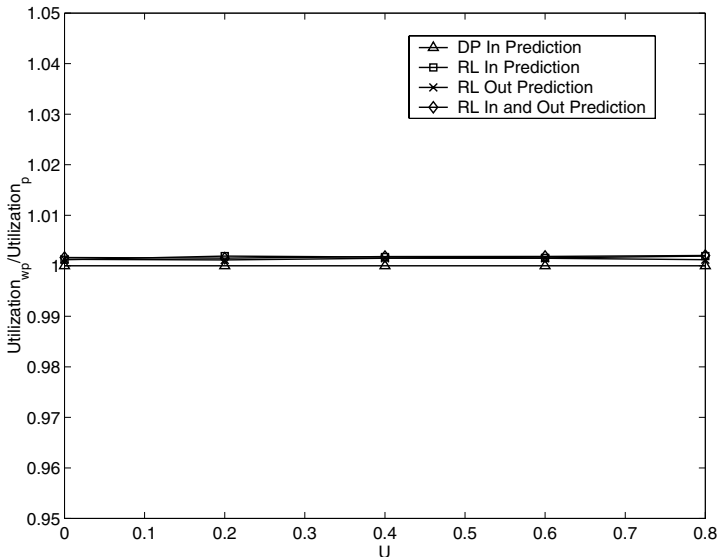


Fig. 6. Utilization gain when using stochastic handover prediction

has also been verified that the resource utilization obtained when deterministic time prediction is deployed is identical to the utilization achieved when the SAC policy is optimized without using predictive information.

As a conclusion, it looks clear that when using optimization techniques to determine the optimum policy, the information related to outgoing handovers is not relevant. This result would seem to contradict the conclusions in [5], but there the predictive information is integrated in the reservation scheme by means of heuristics and therefore their approximation and ours are not comparable.

5 Conclusions

In this paper we evaluate the performance gain that can be expected when the SAC optimization process is provided with information related to incoming, outgoing and incoming and outgoing handovers together, in a mobile cellular network scenario. The prediction information is provided by two types of prediction agents that label active mobile terminals in the cell or its neighborhood which will probably execute a handover. The prediction agents also provide information about the future time instants at which handovers will occur, being this information either stochastic or deterministic. The optimization problem is formulated as a Markov or semi-Markov decision process, for which different solving methods can be used. In this case we deployed dynamic programming and reinforcement learning. A general model of the prediction agents has been considered and as such it cannot be used to obtain concrete results for specific systems nor evaluate the added complexity of deploying a particular prediction method in operational systems. Nevertheless, the generality of the prediction model together with the optimization-based approach permit to obtain bounds for the gain of specific prediction schemes used in conjunction with SAC.

Numerical results show that the information related to the incoming handovers is more relevant than the one related to the outgoing handovers in the optimization framework deployed. Additional performance gain can be obtained when more specific information is provided about the handover time instants, i.e. when their prediction is deterministic instead of stochastic. The gain obtained has been as high as 25% in the studied scenario even when the prediction uncertainty is 20%.

In a future work we will study reinforcement learning algorithms different from SMART, which hopefully will be able to find better solutions in less time, even with more complex state spaces. Another aspect that deserves a closer study is the identification of the parameters that affect the optimum value of T and the study of its sensitivity.

Acknowledgments

This work has been supported by the Spanish Ministry of Education and Science (30% PGE, 70% FEDER) under projects TIC2003-08272 and TEC2004-06437-C05-01, and by the Universidad Politécnic de Valencia under “Programa de Incentivo a la Investigación”.

References

1. 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.
2. N. Bartolini, "Handoff and optimal channel assignment in wireless networks," *Mobile Networks and Applications (MONET)*, vol. 6, no. 6, pp. 511–524, 2001.
3. N. Bartolini and I. Chlamtac, "Call admission control in wireless multimedia networks," in *Proceedings of IEEE PIMRC*, 2002.
4. 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.
5. 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.
6. Roland Zander and Johan M Karlsson, "Predictive and Adaptive Resource Reservation (PARR) for Cellular Networks," *International Journal of Wireless Information Networks*, vol. 11, no. 3, pp. 161-171, 2004.
7. V. Pla, J. M. Giménez-Guzmán, J. Martínez and V. Casares-Giner, "Optimal admission control using handover prediction in mobile cellular networks," in *Proceedings of the 2nd International Working Conference on Performance Modelling and Evaluation of Heterogeneous Networks (HET-NETs '04)*, 2004.
8. M. L. Puterman, *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. John Wiley & Sons, 1994.
9. R. Sutton and A. G. Barto, *Reinforcement Learning*. Cambridge, Massachusetts: The MIT press, 1998.
10. T. K. Das, A. Gosavi, S. Mahadevan, and N. Marchallick, "Solving semi-markov decision problems using average reward reinforcement learning," *Management Science*, vol. 45, no. 4, pp. 560–574, 1999.
11. C. Yener, A. Rose, "Genetic algorithms applied to cellular call admission: local policies," *IEEE Transaction on Vehicular Technology*, vol. 46, no. 1, pp. 72–79, 1997.