Optimal Admission Control in Multimedia Mobile Networks with Handover Prediction

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

Abstract

Admission control is one of the key traffic management mechanisms that must be deployed in order to meet the strict requirements on dependability imposed to the services provided by modern wireless networks. We study the problem of optimizing admission control policies in mobile multimedia cellular networks when predictive information regarding the movement of mobile terminals is available.

For the optimization process we deploy a novel Reinforcement Learning approach based on the concept of afterstates. The results obtained define theoretical limits for the gain that can be expected when using handover prediction, which could not be established by deploying heuristic approaches.

Numerical results show that the performance gain is a function of the anticipation time with which the admission controller knows the occurrence of handovers and an optimal anticipation time exists. We also compare an optimal policy obtained deploying our approach with a previously proposed heuristic prediction scheme, showing that there is still room for technological innovation.

I. Introduction

The widespread success of positioning systems like the global positioning system (GPS) has spurred the research on positioning-based services and mobile tracking techniques. Most handheld GPS receivers are now accurate to within 20 meters or so, but new improvements in GPS technology are expected to increase the accuracy to a 1 meter margin [1]. The European global navigation satellite system Galileo is being developed to be compatible with the GPS system, which will allow an integrated Galileo/GPS receiver to provide the location service simultaneously from both Galileo and GPS satellites with much higher accuracy. It is expected that location services will be greatly enhanced, particularly in urban areas.

As mobile terminals (MTs) integrate positioning systems to provide location services, mobile networks operators can exploit the new functionality to predict the occurrence of handovers and improve in this way the performance of the network. Two additional factors are creating increasing interest in handover prediction. One is the availability of databases that include layout information of roads and cities around the world. The other is the emergence of sophisticated algorithms that make use of layout and positioning information to estimate the movement of MTs with high accuracy.

The authors are with the Dept. of Communications, Universidad Politecnica de Valencia (UPV), 46022 Valencia, Spain.

One of the main challenges in the resource management of wireless networks is the mobility of terminals. Once a new session has been setup it is very difficult to guarantee that resources will be available in the cells visited during the session lifetime unless proper mechanisms are in place. Session Admission Control (SAC) is one of the key traffic management mechanism in mobile multimedia cellular networks that can help to provide certain degree of quality of service (QoS) guarantees and to meet the strict requirements on dependability imposed to the services provided by modern wireless networks. Conceptually, the SAC system makes decisions on when to accept or reject the setup of new sessions or sessions handed over from a neighboring cell. To reserve resources for future requests, the SAC system can reject low priority requests even at times when resources are available.

This paper explores the SAC from a novel optimization approach that exploits the availability of handover prediction information. The results obtained define theoretical limits for the gain that can be expected when handover prediction is used, which could not be established by deploying heuristic approaches. One of the main limitations of applying conventional optimization approaches to the design of SAC policies in multimedia scenarios is the curse of dimensionality of the problem. Solving methods based on dynamic programming coupled with linear programming are not efficient in these cases.

In this paper we explore a novel Reinforcement Learning (RL) approach based on afterstates, which was suggested in [2]. RL is a simulation-based optimization technique in which an agent learns an optimal policy by interacting with an environment, which rewards the agent for each executed action. Compared to conventional RL, the afterstates approach is independent on the number of services involved, achieves better solutions and does it with higher precision [3].

Conventional RL has been successfully applied to optimize the admission control and routing problem in fixed networks [4], the dynamic channel assignment in wireless networks [5], the multi-rate transmission control problem in WCDMA wireless networks [6], and the bandwidth degradation in wireless networks with rate-adaptive multimedia services [7].

The rest of the paper is organized as follows. We first describe the main types of prediction systems and review previously proposed SAC schemes that make use of predictive information. Next, we introduce the theory of Markov Decision Processes and apply this framework to a simple scenario. We then present the basic concepts of RL and apply RL to determine optimal policies when predictive information is available. Finally, we compare our approach to a heuristic one.

II. Prediction Systems

The design of movement prediction schemes and their application to estimate the occurrence of future handovers has been extensively studied in the literature. There are three main types of prediction systems: history-based, location-based and hybrid.

In history-based systems, each base station (BS) collects information about each MT like: previous and next cell visited, sojourn time in the cell, etc. This information can be treated in a personalized way or in an aggregated way. Given that the handover behavior of an MT is statistically similar to that of the MTs arriving to a reference cell

from the same neighboring cell, then collecting the aggregated history of similar movement patterns can improve the accuracy of the prediction.

Examples of history-based prediction systems are [8] and [9]. In [8] a prediction scheme is proposed to determine the *active mobile probabilities*, i.e. the probabilities that a particular MT will be active at future time instants in each of the cells belonging to the set of cells in the vicinity of its current location and along its direction of travel (*shadow cluster*). The authors of [9] propose that BSs record the following quadruplet for each MT: the time when the MT departed from the cell, the previously visited and next visited cells and the sojourn time of the MT in the cell. The first element of the quadruplet is recorded because mobility behavior changes with the time of the day. With this historical information the prediction system described in [9] can estimate, for each MT with an ongoing session, both the probability that the next visited cell is a given one and the probability that a handover will be executed in the near future.

Some problems have been associated with history-based schemes, the most important one being its limited ability to track short-term pattern changes. Positioning-based systems do not suffer from this limitation. They may be characterized as mobile-based, network-based and hybrid. Scalability issues and the success of positioning systems make mobile-based techniques the most commonly deployed. Although pure positioning-based schemes have been proposed in the literature [10], hybrid prediction systems like the one proposed in [11] can substantially improve the prediction accuracy. The system in [11] is a sophisticated positioning-based scheme assisted by road topology information. A database stores information related to road segments like: transition probabilities between neighboring segments (computed from historical information), statistical data of the time taken to transit each segment and statistical data of handovers along each segment. With this information the authors claim that they can manage the fact that actual cell boundaries are fuzzy and irregularly shaped, a feature not addressed by previous proposals.

III. SAC WITH PREDICTIVE INFORMATION

It is usually accepted that it is more disturbing for a subscriber in a cellular network to have an ongoing session dropped than the blocking of a new session setup. To minimize the session dropping or forced termination probability, operators deploy handover prioritization schemes like reserving a number of channels in each cell, named guard channels, only for arriving handovers. Given that as more guard channels are reserved the carried traffic diminishes, it is crucial to dimension its number appropriately. In this sense, schemes that deploy a dynamic number of guard channels, that depend on the momentary conditions of the network, are preferred to static ones. Different schemes proposed in the literature adjust the number of guard channels as a function of the predicted occurrence of handovers. See for example [9], [11] and references therein.

In the scheme proposed in [9], each BS informs its neighbors about the number of BUs (bandwidth units or channels) to be reserved for future handovers. For example, if we denote by BS0 a reference BS, then BS0 sends periodically to its neighbors the length of the prediction window T_{est} to be used. For each ongoing session m a neighboring BSi determines the probability that the session will be handed over to BS0 within T_{est} time units

$p_h(m,i,0)$. Then BSi computes $\sum_m b_m p_h(m,i,0)$ and sends it to BS0, where b_m is the number of BUs occupied by session m. Finally, BS0 aggregates all reservations sent by its neighboring BSs, that we denote by $B_{target}(0)$, and tries to reserve this amount of BUs. The scheme is able to limit the forced termination probability by adapting T_{est}.

One of the novelties of the SAC scheme proposed in [11] is that it 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. Their SAC scheme is based on determining the time instants at which incoming and outgoing handovers will occur within a limited time-window into the future of length T_{thres}. At BS0, the number of BUs that need to be reserved during the next T_{thres} time units, $B_{target}(0)$, are determined by the following algorithm that we describe in a simplified way. Initially set a local variable $x = 0$. Mark the time instants at which incoming and outgoing handovers are predicted to occur in the time-window of length T_{thres}. Proceeding toward the future and until the time-window finishes, increment x each time an incoming handover is encountered by the number of BUs required to carry the arriving session and decrement x each time an outgoing handover is encountered by the number of BUs left free by the leaving session. Let x_{max} be the maximum value of x during the process, i.e. the maximum bandwidth requirement during the prediction window. Then set $B_{target}(0) = x_{max}$. As in [9], T_{thres} is adjusted to limit the forced termination probability.

The admission control at BS0 can be performed in different ways, the simplest one is to accept a new request if after acceptance at least $B_{target}(0)$ BUs remain free. Handover requests are always accepted if enough free resources are available.

IV. Markov Decision Processes

As shown in the preceding section, most of previous studies propose a prediction system and a companion admission control scheme that makes use of the information provided by the former in a heuristic way. Besides, they assume that an admission control policy based on guard channels is the best possible policy. We propose a novel optimization approach based on the formalism of Markov Decision Processes (MDPs) and deploy Reinforcement Learning as the solution method. We determine the optimal (ideally) or quasi-optimal (in practice) admission policy when the admission controller is provided with predictive information.

MDPs can be applied to the study of sequential decision problems when the stochastic behavior of the system can be described as a Markov process. In discrete-time MDPs, when arriving to a new state a decision is made (or an *action* is taken), which is rewarded with some immediate *revenue* or penalized with some immediate *cost*. Those time instants at which decisions are taken are called *decision epochs*. It is clear that actions influence the transition probabilities of the process and therefore its future evolution. We consider stationary deterministic Markovian policies, where the action is influenced only by the current state $\boldsymbol{x}$, i.e. when policy π is followed then action $a = \pi(\boldsymbol{x})$ is chosen in state $\boldsymbol{x}$. In continuous-time MDPs, costs can be more conveniently expressed in terms

of cost rates.

As an example, consider a simple fixed network scenario in which two services contend for the resources of a single link. We make the common assumptions of Poisson arrival processes and exponentially distributed service times. New sessions of service i arrive at rate λ_i, consume b_i BUs and its duration rate is μ_i. Suppose that at blocking states for service i the system incurs in a cost rate of $\lambda_i w_i$ and we want to design an admission policy that minimizes the cost rate in the long-term.

For this type of continuous-time systems and long-term objective it is convenient to formulate the optimization problem as the minimization of the *average cost rate* [12]. The average cost rate, γ^π, is defined as the cost per time unit accrued by the system in the stationary regime when following policy π. In systems like ours, γ^π does not vary with the initial state [12]. We consider the problem of finding the policy π^* that minimizes γ^π, which we name the optimal policy. For the problems we consider, optimal policies always exist. It can be easily shown that the cost structure has been chosen so that the average cost rate represents a weighted sum of loss rates, $\gamma^\pi = \omega_1 P_1 \lambda_1 + \omega_2 P_2 \lambda_2$, where P_i is the blocking probability of service i requests.

For continuous-time MDPs, the Howard equations relate the cost rates and relative values (defined below) of the system when policy π is followed

$$\gamma_{\boldsymbol{x}}(a) - \gamma^\pi + \sum_{\boldsymbol{y} \neq \boldsymbol{x}} q_{\boldsymbol{xy}}(a)(v_{\boldsymbol{y}}(\pi) - v_{\boldsymbol{x}}(\pi)) = 0 \tag{1}$$

where $\gamma_{\boldsymbol{x}}(a)$ is the cost rate at state $\boldsymbol{x}$ when action $a = \pi(\boldsymbol{x})$ is taken, γ^π is the average cost rate, $q_{\boldsymbol{xy}}(a)$ is the transition rate from state $\boldsymbol{x}$ to state $\boldsymbol{y}$ when action $a = \pi(\boldsymbol{x})$ is taken and $v_{\boldsymbol{x}}(\pi)$ is the relative value of state $\boldsymbol{x}$. Clearly, all these terms are influenced by policy π. Intuitively, the term $v_{\boldsymbol{x}}(\pi)$ is the difference between the total cost incurred when the system starts at state $\boldsymbol{x}$ and the total cost incurred if the cost rate at all states were γ^π.

In our example, we consider a link with $C = 5$ BUs, $\boldsymbol{b} = \{1, 2\}$, $\boldsymbol{\lambda} = \{0.3, 0.15\}$ and $\boldsymbol{\mu} = \{0.2, 0.1\}$. The system state vector is defined as $\boldsymbol{x} = (x_1, x_2)$, where x_i is the number of sessions of service i in progress. When the *Complete Sharing* (CS) policy is deployed, then the average cost rates at the system states are: $w_1\lambda_1 + w_2\lambda_2$ for states $\{(5,0), (3,1), (1,2)\}$, $w_2\lambda_2$ for states $\{(4,0), (2,1), (0,2)\}$ and zero for the rest of states. From the Howard equations, γ^π and the relative values $v_{\boldsymbol{x}}(\pi)$ for all states can be determined for this policy. In doing so, observe in (1) that it is not the absolute values of $v_{\boldsymbol{x}}(\pi)$ what are important but their differences. Therefore one can define, for example, $v_{\boldsymbol{0}}(\pi) = 0$ and solve the system of linear equations.

Dynamic Programming (DP) algorithms exploit the fact that the cost functions satisfy the Howard equations. The two most widely known DP algorithms are *policy iteration* and *value iteration*. Both are iterative algorithms that improve a starting policy until the optimum policy is found. For brevity we only describe the policy iteration algorithm. Once γ^π and the set $v_{\boldsymbol{x}}(\pi)$ have been determined for a given policy, the policy can be improved by finding at each state $\boldsymbol{x}$ the action a that minimizes $\{\gamma_{\boldsymbol{x}}(a) - \gamma^\pi + \sum_{\boldsymbol{y} \neq \boldsymbol{x}} q_{\boldsymbol{xy}}(a)(v_{\boldsymbol{y}}(\pi) - v_{\boldsymbol{x}}(\pi))\}$, which is basically equation (1). It can be shown that the new policy is never worse than the previous policy, i.e. its average cost rate is lower or equal than that of the previous policy. The iterative process of policy evaluation and improvement proceeds until γ^π can no longer be improved.

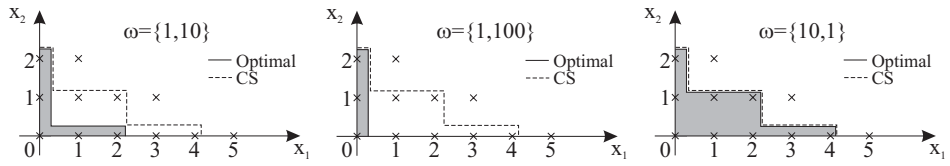
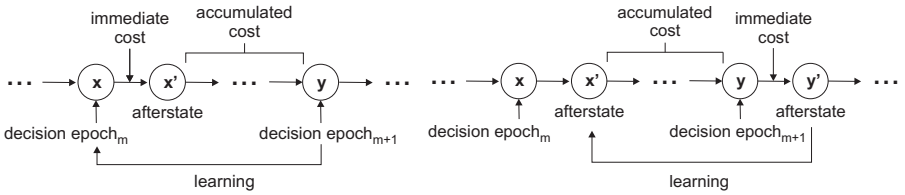


Fig. 1. Optimal policy for service 1.



(a) Conventional RL. (b) Afterstates RL.

Fig. 2. Comparison of the two RL approaches.

Figure 1 shows the optimal policies in three different scenarios that differ in the cost of rejecting a session. In the first two scenarios, requests of service 2 are always accepted while requests of service 1 are accepted only in the states inside the shaded surface. The dotted line shows the acceptance region for service 1 when deploying the CS policy. In the third scenario rejecting a service 1 request is more costly that rejecting a service 2 one. It is not surprising that now the optimal policy is to reject always service 2 requests, given that they occupy more resources and are not that valuable. Therefore, states where $x_2 > 0$ are not reachable. On the other hand, service 1 requests are always accepted while enough free resources are available.

V. Reinforcement Learning

To determine the optimal policy in the previous section, the algorithm requires various iterations and the value of the parameters that describe the system model, like transition rates and average costs, are required. DP algorithms require prohibitive amounts of computation for large state sets. Different methods have been proposed for approximating MDP solutions with less computational effort than required by conventional DP, one of which is Reinforcement Learning. Most RL algorithms adapt standard methods of DP so they can be used in simulation models. Besides, RL algorithms are model free and do not require the value of the system parameters. This might be an advantage in mobile networks where random mobility and fuzzy cell boundaries make it difficult to determine them.

The efficiency of RL algorithms is achieved by combining the following two features [2]. First, they avoid the exhaustive iterations of DP by restricting computation to states on multiple sample trajectories typically obtained by simulation. This allows to concentrate on states which have high probabilities of occurring in a real scenario. And second, at each state, instead of generating and evaluating all of its possible immediate successors, it concentrates on those that look more promising. Figure 2(a) shows how a conventional RL algorithm works. Observe that if state

$\boldsymbol{x}$ is visited at the m-th decision epoch then learning takes place after the occurrence of the next decision epoch.

Intuitively, in systems as the one being considered, afterstates RL is based on the idea that what is relevant in the RL approach is the state reached immediately after the action is taken. More specifically, all states at decision epochs in which the immediate actions drive the system to the same afterstate would accumulate the same future cost. In our simple fixed network example, accepting a service 1 arrival in state $(1,1)$ and accepting a service 2 arrival in state $(2,0)$ would take the system to the same afterstate $(2,1)$. Observe in Fig. 2(b) that if state $\boldsymbol{x}$ is visited at the m-th decision epoch then the learning occurs after the occurrence of next afterstate.

VI. Case Study

In this section we apply the afterstates RL methodology to the design of optimum admission control policies when prediction information is available.

A. System Model

We consider a single cell system and its neighborhood, where the cell has a total of C BUs and the neighborhood C_p BUs, being the physical meaning of a BU dependent on the specific technological implementation of the radio interface. A total of N different services are offered by the system. For each service new and handover session arrivals are distinguished so that there are $2N$ arrival types.

For the sake of mathematical tractability we make the common assumptions of Poisson arrival processes and exponentially distributed random variables: session duration (μ_i^s), cell residence time (μ_i^r), resource holding time ($\mu_i = \mu_i^s + \mu_i^r$) and residence time in the neighborhood (μ_i^p), with rates for service i sessions in parentheses. We assume a circular-shaped cell of radius r and a holed-disk-shaped neighborhood with inner (outer) radius $1.0r$ ($1.5r$). The ratio of arrival rates of new sessions to the cell neighborhood and to the cell is made equal to the ratio of their surfaces. The ratio of handover arrival rates to the cell neighborhood from the outside of the system and from the cell is made equal to the ratio of their perimeters. Without loss of generality, we assume that only one session is active per MT.

B. Prediction System

Given that the focus of our study was not the design of the prediction system, we used a model of it instead. An active MT in the cell neighborhood is labeled by a classifier for incoming handovers 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...). The labeling of the sessions occur T time units before the destination of the MT is definitive. A similar approach is used in [11], where the prediction system determines the time instants at which incoming handovers will take place in a time window of fixed size.

Once the actual destiny of an MT becomes definitive, two outcomes are possible: either a handover into the cell occurs or not (for instance because the session ends or the MT moves to another cell). The label of the MT is

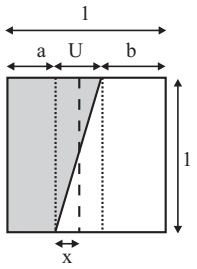


Fig. 3. Classifier model for incoming handovers.

removed either when the MT leaves the neighborhood or when its ongoing call finishes, whichever happens first. 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. 3. It takes into account the inaccuracy of predictions as classification errors, which can be of two types: false-positives and non-detections. Conceptually, the model is depicted by a square with a surface equal to one (1×1), which represents the population of active MTs to be classified. The shaded area (S_H) represents the fraction of MTs that will ultimately move into the cell, while the white area represents the rest of active MTs. 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 are labeled as H and those on the right side as NH. There exists an uncertainty zone, of width U, which accounts for classification errors: the white area on the left of the threshold and the shaded area on the right of the threshold. 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.

C. Numerical Evaluation

The scenario that we consider is characterized by the following parameters: $C = 50$ BUs, $C_p = 100$ BUs, $N = 4$, $\boldsymbol{b} = \{1, 2, 4, 6\}$, $\mu_i = \mu_i^s + \mu_i^r = 1$, $\mu_i^r/\mu_i^s = 1$, $\mu_i^p = 2\mu_i^r$, $S_H = 0.4$ and $x = U/2$. Given that in multiservice wireless networks the bandwidth required by different sessions are quite different, the rate charged per minute is also quite different, and this in turn makes the arrival patterns quite different. The arrival rate of new sessions to the cell for service 1 is set to $\lambda_1^n = 14$. Arrival rates of the other services are set to 20% of the arrival rate of the service with the next lower index. In the scenarios of study the normalized load per BU is 0.62 and the new session blocking probabilities and forced termination probabilities are $P_i^n \approx 10^{-2}$ and $P_i^{ft} \approx 10^{-3}$, respectively.

With regard to the RL algorithm, as no actions are taken at session departures, then only the arrival events are relevant to the optimization process. 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 cost incurred by accepting any arrival type is zero and by rejecting a new request of service i is ω_i^n and a handover request is ω_i^h, where $\boldsymbol{w^n} = \{1, 2, 4, 8\}$ and $\boldsymbol{w^h} = \{20, 40, 80, 160\}$. Note that $\omega_i^n < \omega_i^h$ since the loss of a handover

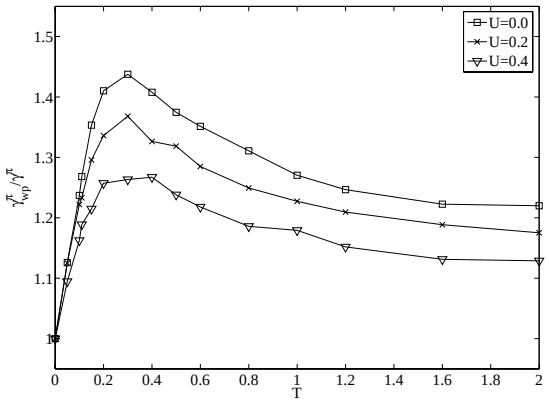


Fig. 4. Performance gain when using handover prediction in a four services scenario.

request is less desirable than the loss of a new session setup request. The average cost rate is defined as a weighted sum of loss rates for the $2N$ arrival types.

The system state is defined by (x_0, x_{in}^w), where x_0 is the number of BUs occupied in the cell under study and x_{in}^{ω} denotes the weighted number of BUs required for the forecasted handover sessions [3]. Note that the weighted number of BUs required for the forecasted handover sessions provides the system with information about their importance (weight). An additional advantage is that with this state representation we obtain a reduced cardinality when compared to a state representation that considers the number of ongoing sessions of each service. Finally, it was shown in [13] that the performance of the policy obtained with this representation is as good as the performance of the optimum policy.

We evaluate the performance gain by the ratio $\gamma_{wp}^{\pi}/\gamma_{p}^{\pi}$, where γ_{p}^{π} (γ_{wp}^{π}) is the average cost rate of the optimal policy in a system with (without) prediction. Figure 4 shows the variation of the gain obtained for different values of T and U. For $U = \{0, 0.2, 0.4\}$, the false-positive probabilities are $\{0, 0.06, 0.12\}$ while the non-detection ones are $\{0, 0.04, 0.08\}$.

As observed, there exists an optimum value for T. As T goes beyond its optimum value the gain decreases because the temporal information becomes less significant for the SAC decision process. When T is lower than its optimum value the gain also decreases because the system has not enough time to react. As an extreme value, when $T = 0$ the gain is null because there is not prediction at all. Due to the simulation-based nature of RL, each point in the figures represents the average of 10 different simulation runs initialized with different seeds. For a confidence level of 95%, the confidence intervals for each point are so narrow that have not been plotted. As shown in Fig. 4, the precision of the prediction subsystem has an impact on the overall performance of the system. Notwithstanding, there already exist sophisticated positioning-based schemes assisted by road topology information, like the one proposed in [11], that can be used to improve the prediction accuracy.

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. It can be shown that the utilization obtained by the policies learned when deploying prediction is practically identical to the one obtained when not deploying prediction, what justifies the efficiency of our optimization approach.

It was shown in [3] that providing the optimization procedure with the additional information of the outgoing handovers was not relevant when deploying the afterstates RL approach. This is due to the fact that the impact of the outgoing handovers is already learned by the RL algorithm and therefore this information is not required to be provided explicitly. The robustness of the afterstates RL approach was evaluated in [13], where it was shown that good policies can be obtained even in non-Markovian scenarios.

To determine *good policies* the learning process requires a generous exploration of the state space, which in turn reduces the convergence rate [3]. The exploration is a common RL technique used to avoid being trapped at local minima. Given that the objective of our study is to determine bounds for the performance gain that can be obtained when using predictive information, it seems logical to deploy a thorough exploration in order to obtain better solutions. Note that the determination of a good policy using a thorough exploration takes only 1 minute in an Intel Core 2 Quad Q6600. Therefore, deploying RL in real operating networks is still possible using historical information, estimating the system parameters periodically and feeding them to the simulation program or a combination of both techniques.

D. Comparative Evaluation

The performance of the SAC policy obtained by the RL optimization approach is compared to the performance of one of the predictive SAC schemes proposed in [9]. Although the schemes proposed in [9] are evaluated in a scenario with two services, the forced termination probability objective is defined for the aggregated of both services. As making a fair comparison in a multiservice scenario is unfeasible, we make it in a single service scenario.

Among the schemes proposed in [9] we chose scheme AC1 instead of AC2 or AC3 because the evaluation scenario deployed so far is better suited for it. Additionally, the performance of the three schemes was evaluated in [11] and the authors concluded that AC1 performs better than the others.

Figure 5 compares the performance of the AC1 scheme with the performance of different policies obtained by the RL approach with $C = 10$ BUs, $\lambda_1^n = 3.5$, $\omega_1^n = 1$ and with the value of the rest of the parameters as defined previously. The value of ω_1^h is conveniently changed in order to obtain different values in the curves of Fig. 5. It is clear that the policies obtained by the RL approach perform better than the AC1 scheme, showing that there is still room for technological innovation.

VII. CONCLUSION

In this paper we evaluated the performance gain that can be expected in a multiservice mobile cellular network scenario when the SAC optimization process is provided with predictive information related to incoming handovers.

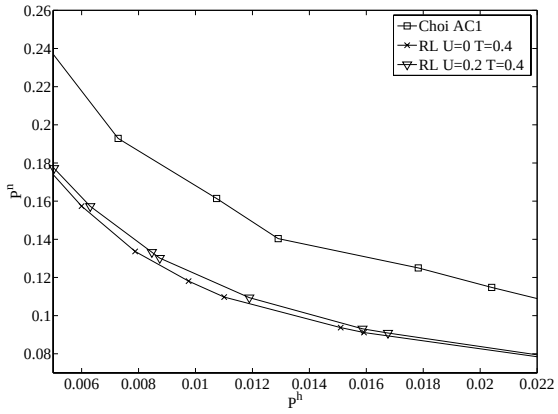


Fig. 5. Blocking probabilities of optimal and heuristic policies.

The prediction system deployed determines the future time instants at which handovers will occur. A classifier labels the active mobile terminals in the neighborhood which will probably execute a handover. The labeling occurs T time units before the handover takes place. The admission controller is aware of the number of labeled mobile terminals at any time.

The optimization problem is solved using a novel Reinforcement Learning algorithm based on the concept of afterstates. With the system state representation used, the computational complexity of the afterstates approach is independent of the number of services being offered by the network.

Finally, we compared our SAC scheme based on optimization with a heuristic one proposed in [9]. The results show that the performance of the policies obtained by our approach is clearly higher.

REFERENCES

[1] E. A. Bretz, "X marks the spot, maybe," IEEE Spectrum, vol. 37, no. 4, pp. 26-36, Apr. 2000.

[2] R. Sutton and A.G. Barto, "Reinforcement Learning," Cambdridge, Massachusetts: The MIT press, 1998.

[3] J.M. Gimenez-Guzman, J. Martinez-Bauset and V. Pla "A Reinforcement Learning approach for admission control in mobile multimedia networks with predictive information," IEICE Trans. on Communications, vol. E90-B, no. 7, pp. 1663-1673, Jul. 2007.

[4] H. Tong and T. X. Brown, "Adaptive call admission control under quality of service constraints: a reinforcement learning solution," IEEE J. Select. Areas Commun., vol. 18, no. 2, pp. 209-221, Feb. 2000.

[5] J. Nie and S. Haykin, "A Q-learning based dynamic channel assignment technique for mobile communication systems," IEEE Trans. Veh. Technol., vol. 48, no. 5, pp. 1676-1687, Sep. 1999.

[6] Y.-S. Chen, C.-J. Chang and F.-C. Ren, "Q-learning-based multirate transmission control scheme for RRM in multimedia WCDMA systems," IEEE Trans. Veh. Technol., vol. 53, no. 1, pp. 38-48, Jan. 2004.

[7] F. Yu, V.W.S. Wong and V.C.M. Leung, "Efficient QoS Provisioning for Adaptive Multimedia in Mobile Communication Networks by Reinforcement Learning," ACM/Springer Mobile Networks and Applications Journal (MONET), vol. 11, no. 1, pp. 101-110, Feb. 2006.

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

[9] S. Choi and K.G. Shin, "Adaptive bandwidth reservation and admission control in QoS-sensitive cellular networks," IEEE Trans. on Parallel and Distributed Systems, vol. 13, no. 9, pp. 882–897, Sep. 2002.

[10] M.-H. Chiu and M.A. Bassiouni, “Predictive schemes for handoff prioritization in cellular networks based on mobile positioning,” IEEE J. Select. Areas Commun., vol. 18, no. 3, pp. 510-522, Mar. 2000.

[11] W.-S. Soh and H.S. Kim, “A predictive bandwidth reservation scheme using mobile positioning and road topology information,” IEEE/ACM Trans. on Networking, vol. 14 no. 5, pp. 1078-1091, Oct. 2006.

[12] S. Mahadevan, “Average reward reinforcement learning: Foundations, algorithms, and empirical results,” Machine Learning (Special Issue on Reinforcement Learning), vol. 22, no. 13, pp. 159-195, Jan./Feb./Mar. 1996.

[13] J.M. Gimenez-Guzman, J. Martinez-Bauset and V. Pla,“Optimal admission control in multimedia mobile networks with movement prediction: a sensitivity analysis,” 3rd EURO-NGI Conference on Next Generation Internet Networks Design and Engineering for Heterogeneity (NGI2007), Trondheim, Norway, May 21-23, 2007.