Call Admission Control for Multimedia Cellular Networks Using Neuro-Dynamic Programming

Sidi-Mohammed Senouci[1], André-Luc Beylot[2], Guy Pujolle[1]

[1] Laboratoire LIP6
Université de Paris VI
8, rue du Capitaine Scott
75015 Paris – France
{Sidi-Mohammed.Senouci, Guy.Pujolle}@lip6.fr
[2] ENSEEIHT - IRIT/TeSA Lab
2, rue C. Camichel - BP7122
F-31071 Toulouse Cedex 7 - France
andre-luc.beylot@enseeiht.fr

Abstract. We consider, in this paper, the call admission control (CAC) problem in a multimedia cellular network that handles several classes of traffic with different resource requirements. The problem is formulated as a Semi-Markov Decision Process (SMDP) problem. It is too complex to allow for an exact solution for this problem, so, we use a real-time neuro-dynamic programming (NDP) [Reinforcement Learning (RL)] algorithm to construct a dynamic call admission control policy. A broad set of experiments shows the robustness of our policies compared to the classical solutions such as Guard Channel

1 Introduction

The increasing demand and rapid growth of mobile communications that will provide reliable voice and data communications has massively grown. The service area in these networks is partitioned into cells. Each cell is assigned a set of channels[1]. As a user moves from one cell to another (handoff), any active call needs to be allocated a channel in the destination cell. If the destination cell has no available channel, the call is aborted. One of the goals of the network designer is to keep the handoff blocking probability low. If this task is simple in a mono-class traffic framework, it is quite complicated in a multi-class context. In a multi-class context it is sometimes preferable to block a call of a less valuable class and to accept another call of a more valuable class.

This paper proposes an alternative approach to solve the call admission control (CAC) in multimedia cellular networks using the experience and knowledge that could be gained during real-time operation of the system. The optimal CAC policy is obtained through a form of reinforcement learning algorithm known as Q-learning

[1] Channels could be frequencies, time slots or codes depending on the radio access technique

[1]. This policy is able to reduce the blocking probability for handoff calls and, also, able to generate higher revenues.

The rest of the paper is organized as follows. After the formulation of the CAC problem as an SMDP in section 2, we detail the two different implementations of Q-Learning algorithm (TQ-CAC and NQ-CAC) that solves this SMDP in section 3. Performance evaluation and numerical results are exposed in section 4. Finally, section 5 summarizes the main contributions of this work.

2 Problem Description

We propose an alternative approach to solving the call admission control problem in a cellular network. The approach is based on the judgment that the CAC can be regarded as a Semi-Markov Decision Process (SMDP), and learning is one of the effective ways to find a solution to this problem [3], [4], [5], [6]. In dynamic programming, we assume that the learner agent exists in an environment described by a set of possible states $S = \{s_1, s_2, ..., s_n\}$. It can perform any of possible actions $A = \{a_1, a_2, ..., a_m\}$ and receives a real-valued reward $r_i = r(s_i, a_i)$ indicating the immediate value of this state-action transition.

For the CAC problem, we identify the system states *s*, the actions *a* and the associated rewards *r* as follows:

1. **States:** We consider two classes of traffic *C*1 and *C*2. But, the ideas in this paper can be extended easily to several classes of traffic as well. We define the state of the system $s=(x,e)$ as:
   - $x=(x_1, x_2)$ where x_1 and x_2 are the number of calls of each class of traffic (*C1* and *C2* respectively) in the cell. We do not take into account the states associated with a call departure because no action needs to be taken.
   - $e \in$ {1 =*arrival of a new C1 call*, 2 =*arrival of new C2 call*, 3 =*arrival of a C1 handoff call*, 4 =*arrival of a C1 handoff call*}
2. **Actions:** Applying an action is to accept or reject the call a∈ {1=*accept*, 0=*reject*}
3. **Rewards:** The reward *r(s,a)* assesses the immediate payoff incurred due to the acceptation of a call in state *s*. We set the reward parameters, as shown in Table 1, for each class of traffic. To prioritize handoff calls, larger reward values have been chosen for handoff calls. $r(s,a) = \begin{cases} \eta_i & \text{if } a = 1 \text{ and } e = e_i \\ 0 & \text{otherwise} \end{cases}$

Table 1. Immediate Rewards

η_1	η_2	η_3	η_4
5	1	50	10

This system constitutes an SMDP with a finite state space S = {(x, e)} and a finite action space *A*={0,1}. To solve this SMDP, a particular learning paradigm has been adopted known as *reinforcement learning (RL)*. There exists a variety of RL algorithms. A particular algorithm that appears to be suitable for the CAC task is called Q-learning [1].

The task of the agent is to learn a policy, $\pi : S \to A$, for selecting its next action $a_t = \pi(s_t)$ based on the current state s_t, that maximizes the long-term revenue/utility. For a policy π, the state-action value $Q^{\pi}(s,a)$ (named Q-value) is the expected discounted reward for executing action a at state s and then following policy π thereafter. The Q-learning process tries to find the optimal Q-values in a recursive manner. The Q-learning rule is

$$Q_{t+1}(s,a) = \begin{cases} Q_t(s,a) + \alpha_t \Delta Q_t(s,a), & if\ s = s_t\ and\ a = a_t \\ Q_t(s,a), & otherwise \end{cases} . \quad (1)$$

$$\text{Where } \Delta Q_t(s,a) = \left\{ r_t + \gamma \max_b [Q_t(s'_t, b)] \right\} - Q_t(s,a) . \quad (2)$$

3 Algorithm Implementation

After the specification of the states, actions and rewards, let us describe the two online implementations of the Q-learning algorithm for solving the CAC problem (TQ-CAC and NQ-CAC). The TQ-CAC uses a lookup table to represent the Q-values. In contrast, the NQ-CAC uses a multi-layer neural network. Function approximators such as neural networks are used when the input space consisting of state-action pairs is large or the input variables are continuous.

When there is a call arrival (new or handoff call), the algorithms determine the action according to

$$a = \arg \max_{a \in A(s) = \{0,1\}} Q^*(s,a) . \quad (3)$$

In particular, (3) implies the following procedures. When a call arrives, the Q-value of accepting the call and the Q-value of rejecting the call are determined. If rejection has the higher value, the call is dropped. Otherwise, the call is accepted.

In these two cases, and to learn the optimal Q-values $Q^*(s,a)$, the value function is updated at each transition from state s to s' under action a for the two algorithms as follows:

1. TQ-CAC: (1) is used to update the appropriate Q-value in the lookup table.
2. NQ-CAC: In this case, ΔQ defined in (2) is served as an error signal which is backpropagated in the *back-propagation (BP)* algorithm [1].

We compare our policies with the greedy policy[2] and with the Guard Channel mechanism [2]. The number of guard channels is determined for each traffic period and each traffic class. The guard channel mechanism will be characterized by a vector s which corresponds to the different thresholds, $s = (s_1, s_2, \cdots, s_K)$, where K is the number of classes of traffic. In the present paper, an exact numerical solution has

[2] Policy that always accepts a call if the capacity constraint will not be violated

been derived. To determine the optimal vector s^*, all the configurations s for which $s_1 \le s_2 \le \cdots \le s_K = N$, where N is the number of channels in the cell, were investigated.

4 Simulation

In order to evaluate the benefits of our call admission control algorithms, we simulate a mobile communication system using a discrete event simulation. We consider a fixed channel assignment (FCA) system with *N=24* channels in each cell. The performance of the algorithms has been evaluated on the basis of the total rewards of the accepted calls (*Total rewards*), the total rewards of the rejected calls (*Total Lost Rewards*), and by measuring the handoff blocking probability.

A set of simulations was carried out, including the cases of traffic load varying, and time-varying traffic load. The experimental results are shown in Fig. 1 through Fig. 3. The results show that the reinforcement learning is a good solution for the call admission control problem. The proposed algorithms are considerably powerful compared to the greedy and to the guard channel schemes. In all cases the lost rewards due to rejection of customers and blocking probability of handoff calls are significantly reduced. The total rewards due to acceptance of customers is also significantly increased.

The Q-values were first learned during a training period with a constant traffic load for both *C*1 and *C*2. The parameters used in the simulation are given in Table 1 and 2.

Table 2. Experimental Parameters

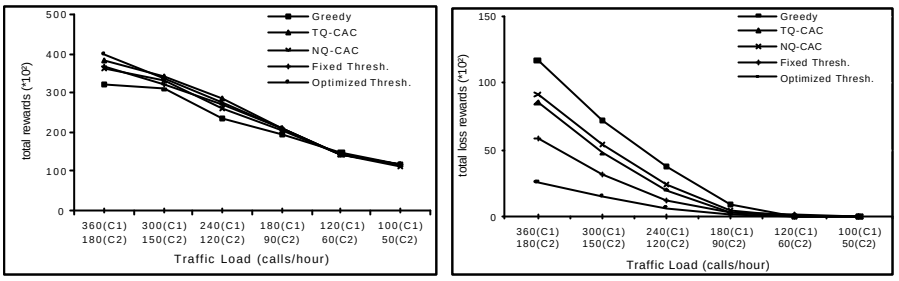
	C1	C2
Number of channels	1	2
Call holding time	40 s	40 s
Call arrival rate	$l_1 = 180\ calls/hour$	$l_2 = l_1/2 = 90\ calls/hour$

4.1 Traffic Load varying

In this case we used the same policy learned in the training period but with six different traffic load conditions (for both classes C1 and C2). Fig. 1 and Fig. 2 show that the proposed algorithms result in significant gains compared with alternative heuristics for all the considered traffic loads and especially when the traffic load is heavy.

It is shown that TQ-CAC leads to significantly better results compared to NQ-CAC because NQ-CAC uses a neural network to represent the Q-values which needs more time to converge.

We also compare our algorithms results to those obtained with: (1) the guard channel with fixed thresholds – these thresholds were calculated for the same traffic load given in Table 2; (2) the guard channel with optimized thresholds - the best thresholds are derived for each input traffic load value.



(a) (b)

Fig. 1. (a) Total rewards per hour (b) Total Loss rewards per hour

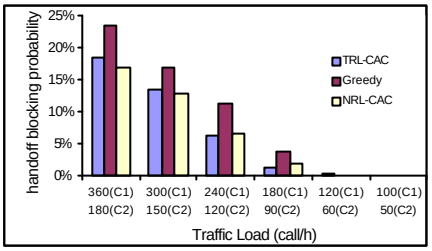


Fig. 2. Handoff blocking with six different traffic loads

This illustrates clearly that TQ-CAC and NQ-CAC have the potential to significantly improve the performance of the system over a broad range of network loads.

We notice in Fig. 1, that the optimal threshold method leads to better performance results than Q-learning. But in this method we must compute the optimal values for each traffic in an off-line manner. In contrast, in TQ-CAC and NQ-CAC, it is interesting to observe that neither the table nor the neural network were relearned and retrained for each traffic load, indicating that the system possesses some generalization and adaptability capabilities.

4.2 Time-Varying Traffic Load

The traffic load in a cellular system is typically time varying. In this case, we use the same policy learned in the training period but during a typical 24-h business day. The peak hours occur at 11:00 a.m. and 4:00 p.m. Fig. 3 gives the simulation results under the assumption that the two traffic classes followed the same time-varying pattern. The blocking probabilities were calculated on an hour-by-hour basis. The improvements of the proposed reinforcement learning algorithms over the greedy policy are apparent specially when the traffic is heavy.

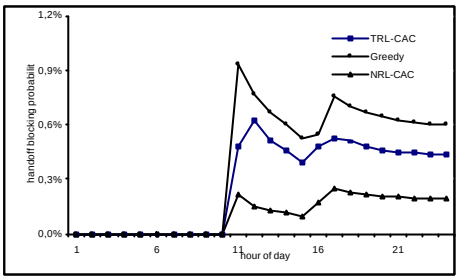


Fig. 3. Performance with time-varying traffic load

5 Conclusion

In this paper, we presented a new approach to solve the problem of call admission control in a cellular multimedia network. We formulate the problem as a dynamic programming problem (SMDP), but with a very large state space. The optimal solutions are obtained by using a self-learning scheme based on Q-Learning algorithm. The benefits gained by this method can be summarized as follows. First, the learning approach provides a simple way to obtain an optimal solution for which an exact solution can be very difficult to find using traditional methods. Second, compared to other schemes like the guard channel, the system offers a generalization capacity. So, any unforeseen event due to significant variations in the environment conditions can be considered as a new experience for improving its adaptation. Third, the acceptation policy can be determined with very little computational effort. It is, also, shown that the proposed CAC algorithms result in significant savings.

References

1. T. M. Mitchell, "Machine Learning", *McGraw-Hill companies, Inc.*, 1997.
2. C.H. Yoon, C.K. Un, Performance of personal portable radio telephone systems with and without guard channels, *IEEE Journal on Selected Areas in Communications (JSAC'1993)*, vol. 11, pp. 911-917, August 1993.
3. P. Marbach, O. Mihatsch and J. N. Tsitsikils, "Call admission control and routing in integrated services networks using neuro-dynamic programming", *IEEE Journal on Selected Areas in Communications (JSAC'2000)*, vol. 18, N°. 2, pp. 197 –208, Feb. 2000.
4. H. Tong and T. X. Brown, "Adaptive Call Admission Control under Quality of Service Constraint: a Reinforcement Learning Solution", *IEEE Journal on Selected Areas in Communications (JSAC'2000)*, vol. 18**,** N°. 2, pp. 209-221, Feb. 2000.
5. R. Ramjee, R. Nagarajan and D. Towsley, "On Optimal Call Admission Control in Cellular Networks", *IEEE INFOCOM*, pp. 43-50, San Francisco, CA, Mar. 1996.
6. S. Senouci, A.-L. Beylot, Guy Pujolle, "A dynamic Q-learning-based call admission control for multimedia cellular networks", IEEE International Conference on Mobile and Wireless Communications Networks (MWCN'2001), pp. 37-43, Recife, Brazil, Aug. 2001.