A Model-Based Q-Learning Scheme for Wireless Channel Allocation with Prioritized Handoff

El-Sayed El-Alfy, Yu-Dong Yao, Harry Heffes

Wireless Systems Engineering Lab,
Dept. of Electrical and Computer Engineering,
Stevens Inst. of Tech., Hoboken, NJ 07030-5991

Abstract–**In this paper we propose a new channel allocation scheme for improving the quality of service in cellular mobile networks. The proposed algorithm prioritizes handoff call requests over new call requests. The goal is to reduce the handoff failures while still making efficient use of the network resources. A performance measure is formed as a weighted linear function of new call and handoff call blocking probabilities. This problem is formulated as a semi-Markov decision process with an average cost criterion. A simulation-based learning algorithm is then developed to approximate the optimal control policy online using the generated samples from direct interactions with the network. It is based on an approximate model that is estimated simultaneously while learning a control policy. The estimated model is used to direct the search for an optimum policy. Extensive simulations are provided to assess the effectiveness of the algorithm under a variety of traffic conditions. Comparisons with some well-known allocation policies are also presented. Simulation results show that for the traffic conditions considered in this paper, the proposed scheme has a comparable performance to the optimal guard channel approach.**

I. INTRODUCTION

Next generation cellular networks are expected to support wideband multimedia applications and wide user mobility anytime and everywhere. One of the faced challenges is to provide quality of service (QoS) guarantees for the generated heterogeneous traffic (voice, video, data, etc) using the limited wireless bandwidth. To increase the system capacity and reduce the power consumption, the coverage area is divided into sub-areas or cells and the available bandwidth units (channels) are assigned to each cell according to some channel assignment strategies [1]-[3] in order to satisfy the channel reuse constraint criterion. The number of channels assigned to each cell base station may be fixed, quasi-fixed, or dynamically changing to accommodate the changes in the traffic within the cell. Within each cell, there are two types of traffic: calls originated in the cell and initially requesting new connections (new calls) and calls migrating from the neighboring cells into the cell (handoffs). Based on the channel allocation criteria, a call (new or handoff) may be denied access to the network resources. From a user's perspective, terminating an ongoing call is more undesirable than blocking an initial call attempt. The impact of the handoff call dropping becomes more serious in the future personal communications networks since there is a trend toward reducing the cell sizes to microcells and even picocells and therefore frequent handoffs are more likely. Consequently, several schemes have been proposed in the literature to reduce the handoff failure probability [1], [4]-[12]. One well-known method is the guard channel approach [4]. The basic idea is to a priori reserve certain number of channels in each cell exclusively to handle handoff requests while the remaining channels are used to serve both types of traffic on a first come first serve basis. In this approach the system trade the new call blocking (and hence the channel utilization) in order to reduce the handoff failures. For that reason, the determination of an optimal admission threshold is critical to maintain a balance between the system utilization and the handoff call dropping probability. In [5], it has been proved that such a threshold exists for which a linear weighted function of new call and handoff blocking probabilities is minimized. Other techniques allow queueing with the guard channel approach of the originated calls [6] and/or the handoff requests [1], [4], [7] until a channel becomes available. If no channel becomes free before a maximum allowable delay, the call is dropped. A trade-off between the blocking probabilities and the increased delay arises. But for microcellular systems there may be several handoffs during a call session duration and queueing, for handoff requests, becomes less favorable.

The exact solution for the optimum threshold is determined in [8] using an analytical model and an incremental search based on the properties of handoff and new call blocking probabilities. Other approaches based on the Markov decision theory [13] find the optimal threshold using synchronous dynamic programming (DP) techniques [10]. All these approaches presume a priori knowledge of a perfect system model. Further, the computational complexity precludes them to scale well for large state-space systems. By allowing the guard threshold to change dynamically to adapt to the traffic conditions, further potential improvements can be attained [11], [12]. There is also a class of algorithms that predict the mobility patterns and reserve the required bandwidth in advance in the destination cells [9], [11].

To reduce the computational burden of the conventional DP techniques, asynchronous value iteration for real and non-real time systems have been suggested [14]. Also, an adaptive real-time dynamic programming scheme (ARTDP) [14] has been proposed for discrete-time Markov decision problems (MDP) in the discounted framework. Another scheme called H-learning [15] has been devised for the average cost counterpart. Recently a model-free reinforcement learning (RL), a stochastic approximation of dynamic programming,

0-7803-7206-9/01/$17.00 © 2001 IEEE

has been applied to the channel assignment problem [2]-[3]. However in these schemes, there is no differentiation between new calls and handoffs. In [16], a class of model-free reinforcement learning schemes is applied for the handoff prioritized call admission control problem and promising results are attained.

In this paper we propose a new channel allocation scheme that gives higher priority to handoffs. The proposed algorithm lowers the handoff dropping probability while maintaining the channel utilization and autonomously adjusts the allocation policy to the traffic scenario. Furthermore, it does not presume a priori knowledge of the system model or the traffic parameters. However, they approximate a model online using the observed sample data collected from direct interaction with the network at the same time they learn a control policy. The estimated model is used to direct the search for an optimal control policy. Intensive simulations are conducted to evaluate the performance of the proposed algorithm against other resource allocation policies, such as the complete sharing and the optimal guard channel policies.

The remainder of this paper is organized as follows. The next section describes the traffic model and the performance measures. In Section III we formulate the channel allocation problem as an average-cost semi-Markov decision process (AC-SMDP). Section IV presents the proposed model-based reinforcement learning scheme. Simulations and numerical results are provided in Section V. Analytical and empirical comparisons with complete sharing and optimal guard channel policies are also given. Finally, in Section VI we present our conclusions.

II. Traffic model and performance criteria

Consider a generic cellular network with a limited number of bandwidth units or channels which may be time slots, frequency carriers, or spreading codes based on the access technology used. There are two kinds of traffic arrivals in each cell based on the call origination location: new and handoff calls. Based on the availability of the system resources and the allocation criteria, a channel allocation scheme decides at each arrival instance whether to accept or reject the call. Under a fixed channel assignment scheme and the assumption of spatially uniform traffic conditions, the cellular network can be studied by focusing on a single cell. Figure 1 shows the traffic model for a particular cell with a fixed number of channels, *C*, as an *M*/*M*/*C* Erlang-loss model. We assume that blocked calls are cleared. The bandwidth requirement can be expressed in terms of bandwidth units (BU's) or channels. We make the commonly used assumptions that the arrivals of new calls and handoff calls are according to mutually independent Poisson processes with mean arrival rates λ_n and λ_h respectively. The call session duration, T_s, and the inter-handoff time (cell dwelling time), T_h, are mutually independent and exponentially distributed with means $1/\mu_s$ and $1/\mu_h$ respectively. Therefore, the channel holding time, the minimum of the call duration and time until handoff, is also exponential with mean $1/\mu = 1/(\mu_s + \mu_h)$. The ratio μ_h/μ_s indicates the relative motion of the mobile station to the cell size and can be referred as mobility index.

The network state $n(t)$ can be defined as the number of busy channels at time t, and the natural evolution of the stochastic process $\{n(t),\ t \geq 0\}$ represents a continuous-time Markov chain with a finite state space $S = \{0, 1, 2, \ldots, C\}$. The transition rate diagram for a general allocation policy is shown in Fig. 2. The aim of any allocation control strategy is to determine the allocation probabilities β_{ni} and β_{hi} for all states. For complete sharing $\beta_{ni} = \beta_{hi} = 1$ for i = 0, 1, 2, …, C-1 and zeros otherwise. Let P_i be the steady state probability that the system occupies state i. Hence, the new call blocking and the handoff call dropping probabilities are the same, that is, $B_n = B_h = P_C$. On the other hand, for the guard channel approach the new call and the handoff admission policies are different; $\beta_{hi} = 1$ for i = 0, 1, 2, …, C-1 and zero otherwise, while $\beta_{ni} = 1$ for i = 0, 1, 2, …, G-1 and zero otherwise where G is a guard threshold value. The blocking probabilities are given by $B_n = \sum_{i=G}^{C} P_i$, and $B_h = P_C$.

The goal is to find a channel allocation policy that minimizes a weighted linear function of new call and handoff call blocking probabilities as defined by

$$P = w_n \frac{\lambda_n}{\lambda_n + \lambda_h} B_n + w_h \frac{\lambda_h}{\lambda_n + \lambda_h} B_h , \qquad (1)$$

where w_n and w_h represent the relative weights of each type with $w_h > w_n$ to reflect the fact that rejecting a handoff request is more undesirable than rejecting a new call request.

In the next section, this problem is formulated as an average cost semi-Markov decision process.

III. The channel allocation problem as an AC-SMDP

The channel allocation problem can be formulated as an infinite-horizon finite-state semi-Markov decision process (SMDP) under the average cost criterion. In the following a reduced state-space model is used to find an allocation policy by controlling the admission of new calls only since it is always optimal to accept a handoff request as long as there is a free channel. The primary components of an AC-SMDP are defined as follow. The *decision epochs* correspond to the new call arrival instances. The *sojourn time* from one decision epoch to the next decision epoch is a continuous time random variable with the same probability distribution as the new call inter-arrival times. The *system state* is defined as the number of busy channels immediately prior to a new call arrival. Although the system state may change between two decision epochs due to handoff call arrivals or calls leaving the cell, only the states at the decision epochs are significant to the controller. The system *state space* is a finite set $S = \{0, 1, 2, \ldots, C\}$. The *action set* available in each state is a finite set $A_s = \{0 = \text{reject}, 1 = \text{admit}\}$ for $s \in \{0, 1, 2, ..., C\text{-}1\}$ and $A_s = \{0$

= reject} for $s\in\{C\}$. A deterministic stationary *policy* is a mapping from states to actions $\pi : S \to A$. Starting from initial state $s_0 = i$ and implementing policy π, the system state at nth decision epoch, s_n, evolves as a semi-Markov chain $\{s_n$: $n \ge 0$, $s_n \in S\}$ with a *state transition probability* $P(s_{n+1} = j|\, s_n = i, a = \pi_i) = P_{ij}^a$, that is, the probability that the system state at the next decision epoch is j given that the current state is i and action π_i is applied. At each decision epoch the controller incurs a random stage cost which depends on the rejected calls on that stage. The immediate stage cost, c_{k+1}, and the sojourn time, τ_{k+1}, are not known until the next decision epoch. The sequence of incurred costs is a stochastic process and depends on the adopted policy. For this ergodic Markov decision process, the average cost is independent of the initial state and is defined as

$$g^{\pi} = \lim_{n\to\infty} E\{\sum_{k=1}^{n} c_k\} \Big/ E\{\sum_{k=1}^{n} \tau_k\}, \qquad \forall \pi \in \Pi. \tag{2}$$

Here Π is a set of all feasible policies. The controller objective is to determine a policy π^* with a corresponding minimum long-term average, i.e., $\pi^* = \arg\min_{\pi\in\Pi} g^{\pi}$.

The Bellman optimality equations [13] for this average cost ergodic Markov decision process have the recurrence form

$$h_x^* = \min_{a\in A_x}\left\{ c_x^a - g^*\tau_x^a + \sum_{y\in S} P_{xy}^a h_y^* \right\} \ \forall x \in S, \tag{3}$$

where h_x^* is an optimal state dependent value function $h^* : S \to \Re$. The corresponding optimal policy is

$$\pi_x^* = \arg\min_{a\in A_x}\left\{ c_x^a - g^*\tau_x^a + \sum_{y\in S} P_{xy}^a h_y^* \right\} \forall x \in S. \tag{4}$$

If a perfect model can be analytically derived; then, the solution of the set of equations (3) and the corresponding policy (4) can be obtained through dynamic programming techniques. In the next section, we present a new paradigm that integrates the model estimation and the policy learning for approximating the optimal solution of (3) online.

IV. The Proposed Model-Based RL Solution

The model-based learning architecture, as illustrated in Fig. 3, has three main subsystems: the controlled system, the control policy learner, and the model estimator. The controlled system could be a real world or a simulated system. The control policy learner has two components: a value function learner and a policy finder.

A. *Model Estimation*

The control system learns a model for the system dynamics from the observed sample values. It estimates the state transition probabilities, the sojourn time until the next decision and the expected immediate cost functions using the sample averages. Let $N_{xy}^a(k)$ be the number of times the system state changes from x to y under action a before the kth decision epoch; $N_x^a(k)$ be the number of times of executing action a in state x before the kth epoch. Then, the controlled state transition probabilities at the kth decision epoch are estimated as follows

$$P_{xy}^a(k) = \begin{cases} \dfrac{N_{xy}^a(k)}{N_x^a(k)} & \text{if } x = x_{k-1}, y = x_k \wedge a = a_{k-1} \\ P_{xy}^a(k-1) & \text{otherwise} \end{cases}, \tag{5}$$

and the average immediate cost is estimated as

$$c_x^a(k) = \begin{cases} c_x^a(k-1) + \dfrac{c_k - c_x^a(k-1)}{N_x^a(k)} & \text{if } x = x_{k-1} \wedge a = a_{k-1} \\ c_x^a(k-1) & \text{otherwise} \end{cases}. \tag{6}$$

Similarly, the average sojourn time is given by

$$\tau_x^a(k) = \begin{cases} \tau_x^a(k-1) + \dfrac{\tau_k - \tau_x^a(k-1)}{N_x^a(k)} & \text{if } x = x_{k-1} \wedge a = a_{k-1} \\ \tau_x^a(k-1) & \text{otherwise} \end{cases}. \tag{7}$$

Under the assumption that every state-action pair is visited infinitely often, the estimated model converges asymptotically (a.s.) to the true model.

B. *Learning the Optimal Value Functions*

In this subsection, we develop a gradient-like scheme imAQ (Incremental Model-based Average-cost Q-learning) to incrementally update the action value functions instead of the state value function in (3). This allows a relatively quick action selection. In contrast to [17], the proposed scheme is for the continuous-time MDP and uses the estimated model instead of the temporal difference to direct the search operation. Following [17], the optimal Q-functions are defined for the AC-SMDP as

$$Q_x^a = c_x^a - g^*\tau_x^a + \sum_{y\in S} P_{xy}^a h^*(y) \ \forall x \in S, \text{and } a \in A_x \tag{8}$$

where Q_x^a represents the value of applying action a in state x and following the optimal policy thereafter. The relationship between the Q-values and h-values are given by

$$h_x^* = \min_{a\in A_x} Q_x^a \ \ \forall x \in S. \tag{9}$$

Hence, equation (8) becomes

$$Q_x^a = c_x^a - g^*\tau_x^a + \sum_{y\in S} P_{xy}^a \min_{b\in A_y}[Q_y^b], \forall x \in S \text{ and } a \in A_x, \tag{10}$$

To allow the determination of Q-values online, the controller replaces the expected values in (10) with the estimated values in (5)-(7) based on the certainty equivalence principle [14]. A more efficient approach is to use a gradient-like scheme that incrementally updates the Q-values using

$$Q_x^a(k) = Q_x^a(k-1) + \alpha\{c_x^a(k) - \rho_k \tau_x^a(k) + \sum_{y\in S} P_{xy}^a(k) \min_{b\in A_y}[Q_y^b(k-1)] - Q_x^a(k-1)\}, \tag{11}$$

where ρ_k is an estimate of the long-term average cost and $\alpha \in (0, 1]$ is a learning rate or a step-size parameter. The setting for α can be fixed or gradually decaying over time. The average cost is estimated using the accumulated costs and sojourn times as follows

$$\rho_k = \sum_{t=1}^{k} c_t I_{t-1} \Big/ \sum_{t=1}^{k} \tau_t I_{t-1}, \tag{12}$$

where I_t, an indicator function, equals one if a greedy action is applied at the t^{th} decision epoch and zero otherwise. To allow faster convergence, the controller can use the estimated model and apply the update rule (11) to more than one state-action pair depending on the available time before the next decision epoch.

Other approaches are still possible for integrating the model-free and the model-based learning approaches. For instance, the model-free schemes are more efficient at the earlier stages and can be used until gaining enough information and the error in the approximate model becomes low; then the policy learner can switch to the model-based approach. But determining the switch point is not a trivial task.

C. The Policy Finder

When a new call arrives, the controller observes the state of the network and determines whether to admit or reject based on the estimated state-action value functions. Various approaches can be used to select an action using the learned action value functions. One simple approach, called greedy selection mechanism, is to always select the one that is apparently the best current action, i.e. the one that has the smallest Q-value. To resolve the conflict between the system identification (exploration) and control (exploitation), more complicated selection mechanisms are needed [18]. One commonly used approach called ε-greedy in which the greedy action is selected with high probability, 1-ε, and with small probability, ε, uniformly select among actions; where ε may be fixed or gradually diminishing over time. Other approaches direct the action selection based on their Q-values, e.g., the Boltzmann's distribution where the exploration rate is controlled through the temperature parameter.

V. Simulation and Numerical Results

In this section, we conduct intensive computer simulation runs to empirically evaluate and compare the performance of the learning algorithm (imAQ) with the complete sharing (CS) and the optimal guard reservation (GC) policies for different traffic scenarios. A single cell with $C = 20$ channels is considered. We built a discrete event simulator to generate the traffic streams for new call and handoff call requests according to mutually independent Poisson processes. In the first experiment, the mean arrival rates are set to $\lambda_n = 5$ calls/min, and $\lambda_h = 3$ calls/min respectively. Each call requests one channel. The channel holding time is exponentially distributed with mean $1/\mu = 2$ min. The weighting factors are set to $w_n = 1$, and $w_h = 10$. The analytical solution reveals that the optimal threshold $G_o = 18$ and the corresponding blocking probabilities $B_h = 0.010128$ and $B_n = 0.150796$. We run the simulator for 10 runs and the average values are depicted in Figs. 4.a-4.c. Fig. 4.a shows the average cost incurred by each policy while Figs. 4.b and 4.c show the corresponding new and handoff call blocking probabilities for each policy. For the learning algorithm, the value functions are initialized to zeros and the step-size parameter α is set to 0.01. The simulation results show that the learning approach was capable of self-adjusting and prioritizing the handoff call. Also, the performance of imAQ is superior to the complete sharing policy and is very close to the optimal guard policy. To test the performance of the learning approach compared to other policies for different traffic conditions (e.g. when the handoff rate is varied), we run the simulator for all the policies for the same settings as in the first experiment but for different handoff rates. The resulting values are averaged over 10 runs. The average values at the end of the simulation time for the average cost incurred, new call blocking probability and handoff call blocking probability are plotted versus the handoff rate in Figs. 4.d-4.f; where CS_A and CS_S refer to the analytical and simulation results for complete sharing policy respectively. Similarly, GC_A and GC_S indicate the analytical and simulation results for guard channel policy respectively. Again as depicted in Fig. 4.d, the complete sharing policy has the highest incurred average cost while the optimal guard channel policy incurred the smallest average cost. The imAQ learning approach has a comparative performance to the optimal guard threshold policy.

VI. Conclusions

In this paper we have addressed the channel allocation in cellular mobile networks with prioritized handoffs. The problem has been formulated as an average cost continuous-time Markov decision problem. Then, a model-based reinforcement learning approach has been developed for finding a self-adjusting allocation strategy that is approximately optimal. Simulation results show that the proposed scheme outperforms the complete sharing policy and has a comparable performance to the optimal guard channel approach. Although in this paper we assumed only handoff prioritization for a single traffic class, the proposed algorithm can easily be extended to deal with multiple traffic classes with heterogeneous characteristics.

REFERENCES

[1] S. Tekinay and B. Jabbari, "Handover and channel assignment in mobile cellular networks," *IEEE Commun. Mag.*, vol. 29, pp. 42-46, 1991.

[2] S. P. Singh and D. P. Bertsekas, "Reinforcement learning for dynamic channel allocation in cellular telephone systems," In *Advances in NIPS 9*, MIT Press, pp.974-980, 1997.

[3] J. Nie and S. Haykin, "A *Q*-learning based dynamic channel assignment technique for mobile communication systems," *IEEE Trans. Veh. Tech.*, 1997.

[4] D. Hong and S. S. Rappaport, "Traffic model and performance analysis for cellular mobile radio telephone systems with prioritized and nonprioritized handoff procedures," *IEEE Trans. Veh. Tech.*, vol. 35, pp. 77-92, 1986.

[5] R. Ramjee, D. Towsley and R. Nagarajan, "On optimal call admission control in cellular networks," *Wireless Networks*, vol. 3, pp. 29-41, 1997.

[6] R. Guerin, "Queueing-blocking system with two arrival streams and guard Channels," *IEEE Trans. on Commun.*, vol. 36, pp. 153-163,1988.

[7] C. Chang, T. Su and Y. Chiang, "Analysis of a cutoff priority cellular radio system with finite queueing and reneging/dropping," *IEEE/ACM Trans. On Networking*, vol. 2, no. 2, pp. 166-175, 1994.

[8] S. Oh and D. Tcha, "Prioritized channel assignment in a cellular radio networks," *IEEE Trans. on Commun.*, vol. 40, pp. 1259-1269, 1992.

[9] W. Su and M. Gerla, "Bandwidth allocation strategies for wireless ATM networks using predictive reservation**,"** *IEEE Global Telecommunications Conference*, GLOBECOM 1998, vol. 4, pp. 2245-2250, 1998.

[10] M. Saquib and B. Yates,"Optimal call admission to a mobile cellular network," *IEEE Veh. Tech. Conf.*, pp. 190-194, 1995.

[11] S. Boumerdassi and A. Beylot, "Adaptive channel allocation for wireless PCN," *Mobile Networks and Applications*, vol. 4, pp. 111-116, 1999.

[12] O. Yu and V. Leung, "Adaptive resource allocation for prioritized call admission over an ATM-based wireless PCN," *IEEE JSAC*, vol. 15, pp.1208-1225, 1997.

[13] M. L. Putterman, *Markov decision processes: Discrete stochastic dynamic programming*. Wiley, New York, 1994.

[14] A. G. Barto, S. J. Bradtke, and S. P. Singh, "Real-time learning and control using asynchronous dynamic programming," *Technical Report 91-57*, Department of Computer Science, UMASS, 1991.

[15] P. Tadepalli and D. Ok "Model-based average reward reinforcement learning," *Artificial Intelligence*, vol. 100, pp.177-224, 1998.

[16] E. El-Alfy, Y. D. Yao, and H. Heffes, "Autonomous call admission control with prioritized handoff in cellular networks," *IEEE International Conference on Communications*, ICC'01.

[17] S. Singh, "Reinforcement learning algorithms for average-payoff Markovian decision processes," *In Proceedings of the 12^{th} AAAI*, 1994.

[18] S. Thrun, "Efficient exploration in reinforcement learning," *Technical Report* CMU-CS-92-102, CMU, School of Computer Science, 1992.

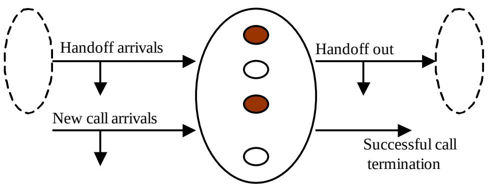


Fig. 1. Cellular system and traffic model.

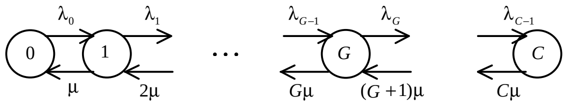


Fig. 2. Transition rate diagram for a generic allocation policy, $\lambda_i = \lambda_n \beta_{ni} + \lambda_h \beta_{hi}$ for $i \in \{0, 1, 2, \ldots, C\text{-}1\}$ and $\mu=\mu_s+\mu_h$.

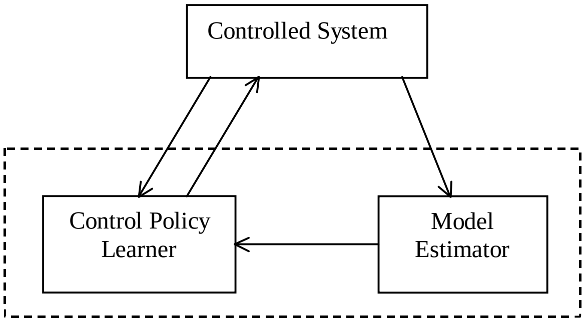
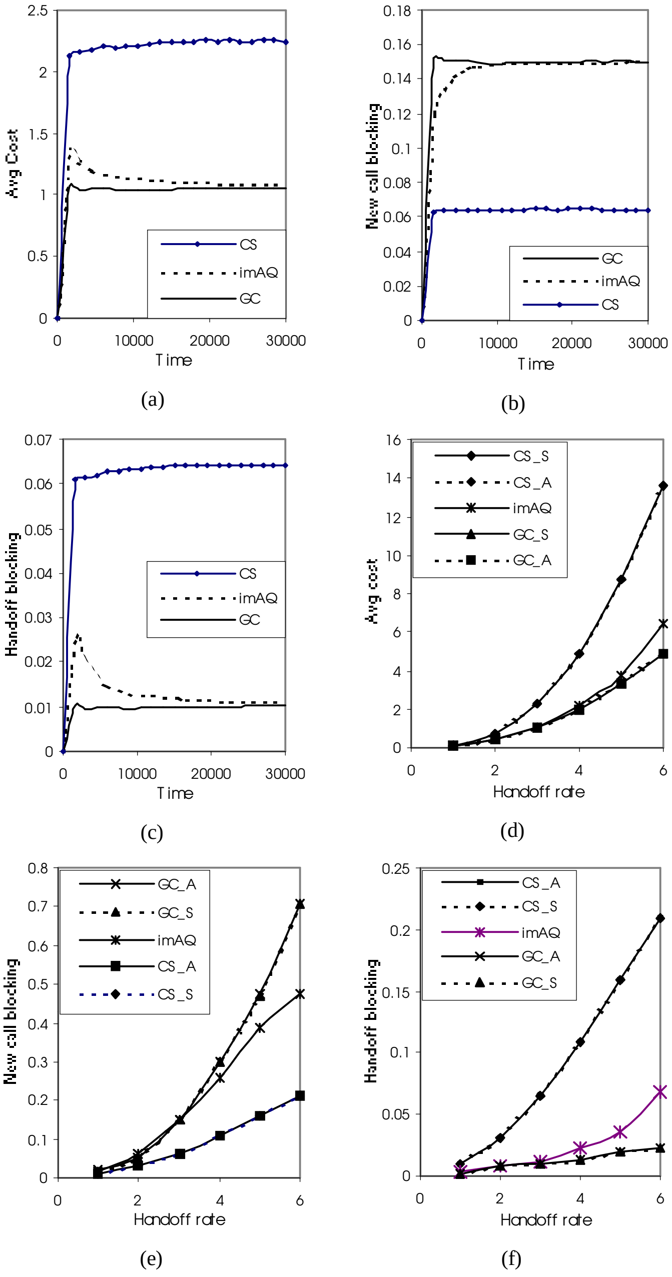


Fig. 3. Model-based system components.



(a)

(b)

(c)

(d)

(e)

(f)

Fig. 4. The performance of imAQ compared with GCP and CSP.