Comparing a class of dynamic model-based reinforcement learning schemes for handoff prioritization in mobile communication networks

El-Sayed M. El-Alfy \textsuperscript{a,}\textsuperscript{*}, Yu-Dong Yao \textsuperscript{b}

\textsuperscript{a} College of Computer Sciences and Engineering, King Fahd University of Petroleum and Minerals, Dhahran 31261, Saudi Arabia
\textsuperscript{b} Wireless Information Systems Engineering Laboratory (WISELAB), Dept. of Electrical and Computer Engineering, Stevens Institute of Technology, Hoboken, NJ 07030-5991, United States

\textbf{A R T I C L E I N F O}

Keywords:
Resource management
Handoff prioritization
Cellular systems
Mobile communication networks
Reinforcement learning
Semi-Markov decision process

\textbf{A B S T R A C T}

This paper presents and compares three model-based reinforcement learning schemes for admission policy with handoff prioritization in mobile communication networks. The goal is to reduce the handoff failures while making efficient use of the wireless network resources. A performance measure is formed as a weighted linear function of the blocking probability of new connection requests and the handoff failure probability. Then, the problem is formulated as a semi-Markov decision process with an average cost criterion and a simulation-based learning algorithm is developed to approximate the optimal control policy. The proposed schemes are driven by a dynamic model estimated simultaneously while learning the control policy using samples generated from direct interactions with the network. Extensive simulations are provided to assess and compare their effectiveness of the algorithm under a variety of traffic conditions with some well-known policies.

\textsuperscript{\textdegree} 2011 Elsevier Ltd. All rights reserved.

1. Introduction

Due to the advances in wireless technologies, mobile communication networks have become a dominating infrastructure for all types of communications nowadays. Promising features of fourth generation wireless networks are the support for wideband multimedia applications, seamless user mobility across heterogeneous networks and ubiquitous coverage (Hussain, Hamid, \& Khattak, 2006; Kouvetos \& Li, 2005). As new and innovative multimedia applications emerge, providing acceptable levels of quality-of-service (QoS) for the diverse traffic (voice, video, and data) in such environments with the limited wireless bandwidth becomes a stringent problem. An important quality measure is the continuation of active sessions while increasing the wireless bandwidth utilization. As the mobile station moves away from one wireless system to another, the received signal power deteriorates, the interference increases and the quality degrades. Hence, a handoff request will be initiated either by the mobile station (mobile-controlled handoff) or by the network (network-controlled handoff). Based on the channel availability and the admission control criteria, the handoff request may be rejected and hence the session is forced to terminate. From a user’s perspective, terminating an ongoing session is more undesirable than blocking an initial session attempt, especially as handoff becomes more frequent.

In this paper we develop a new class of model-based reinforcement learning schemes that inherently give higher priority to handoff requests over new connection requests. The proposed schemes lower the handoff failure probability while maintaining the channel utilization and seamlessly adapt the bandwidth allocation policy to the traffic scenario. Furthermore, they do not presume a priori knowledge of the system model or traffic parameters. However, they approximate a model online using the observed sample data collected from the interaction with the network at the same time as they learn a control policy. The estimated model is used to direct the search for an optimal control policy. Extensive simulations are conducted to evaluate the performance of the proposed schemes. Comparisons with other bandwidth allocation policies, such as the complete sharing and optimal guard channel, are also provided.

The remainder of this paper is organized as follows. The next section presents a brief background and review some related work. Section 3 describes the problem and performance measures. In Section 4, we formulate the problem as an average cost semi-Markov decision process (AC-SMDP). Section 5 presents three model-based reinforcement learning schemes for solving the problem. Simulations and numerical results are provided in Section 6. Analytical and empirical comparisons with complete sharing and optimal guard channel policies are also given. Finally, Section 7 presents the conclusions and future work.
2. Background and related work

In the literature, several approaches have been proposed for channel assignment in wireless cellular networks (Katzela & Naghsineh, 1996; Lee, Park, & Seo, 2008, chap. 8; Oh & Tcha, 1992; Stuber, 2001, chap. 13). Also the handoff has attracted the attention of many researchers and is still an active area of research. It has been addressed in a number of papers under a variety of goals. The research work related to handoff can be divided into three categories: handoff initiation, handoff resource allocation, and connection transfer protocols (El-Alfy, 2001; Huang, Chuang, & Yang, 2008; Huang, Hu, Chen, Chen, & Chen, 2010; Stojmenović, 2002; Tekinay & Jabbari, 1991). Handoff initiation focuses on taking measurements of the link quality (such as the relative signal strength) and decides whether to start handoff or not. Other network layer metrics can be considered at this stage such as bit-error rate, congestion and traffic load; in this case it is known as cross-layer handoff. During the initiation stage, it is required to avoid unnecessary handoffs and reduce handoff latencies (Lin, Wang, & Lin, 2008). Handoff resource allocation is a planning process that ensures the availability of wireless resources in the target base station should a handoff is initiated. Finally connection transfer protocols are responsible for changing the serving base station.

The main idea in various papers for handoff prioritization is to reserve a number of wireless resources (guard channels) exclusively to handle handoff requests while the remaining resources are used to serve both types of traffic (handoff and new connection requests) on a first-come-first-serve basis. The number of the reserved channels can be fixed or dynamically adjusted (Diederich & Zitterbart, 2005; Guerin, 1988; Ramjee, Towsley, & Nagarajan, 1997; Salamah & Lababidi, 2005; Zheng, 2008). With this approach, the system compromises the blocking of new connection requests (and hence the channel utilization) in order to reduce the handoff failures. For this reason, the determination of an optimal policy is critical to maintain a balance between the system utilization and the handoff failure probability. Other techniques allow queuing with the guard channel approach of the new connection requests and/or handoff requests (Chang, Su, & Chiang, 1994; Hong & Rapaport, 1986; Salih & Fidanboylu, 2006; Tekinay & Jabbari, 1991) until a channel becomes available. If no channel becomes free before a maximum allowable delay, the request is eventually dropped. A trade-off between the blocking probabilities and the increased latency arises and needs to be resolved. Hence in some environments, e.g. microcellular systems, Handoff queuing is less favorable.

Other approaches based on the Markov decision theory find the optimal threshold using dynamic programming (DP) synchronous value iteration (Puterman, 1994; Saquib & Yates, 1995) or modified linear programming (LP) (Ho & Lea, 1999) techniques. All these approaches are of limited use and only useful when complete knowledge of the system is known a priori. Further, the computational complexity precludes such techniques to scale well for large state-space systems. In dynamic networks, by allowing the guard threshold to change dynamically to adapt to the traffic conditions, further potential improvements can be attained (Boumerdassi & Beylot, 1999). Mobility and traffic patterns have been used to predict handoff and reserve resources accordingly in candidate cells (Chiu & Bassiouni, 2000; Lu, Wu, & Liu, 2006; Soh & Kim, 2006; Su & Gerla, 1998; Ye et al., 2007). Reinforcement learning (Sutton & Barto, 1998) has been applied for channel allocation to different cells in cellular telephone systems in Singh and Bertsekas (1997). In 1999, a Q-learning based method has been proposed for dynamic channel assignment in mobile communication systems but without differentiation between new connection requests and handoffs (Nie & Haykin, 1999). In our previous work we have applied a model-free reinforcement learning schemes for the handoff prioritized call admission control problem and promising results were attained (El-Alfy, Yao, & Heffes, 2006). Discounted reward reinforcement learning has also been applied to solve the QoS provisioning for adaptive multimedia in mobile communication networks where the bandwidth allocated to a multimedia call is dynamically changed to adapt to traffic fluctuations (Yu, Wong, & Leung, 2006). Another recent application of Q-learning is presented in Song and Jamalipour (2008) for a negotiation based vertical handoff decision scheme for the switching process in heterogeneous wireless networks.

3. Problem description

Consider a cellular network with a limited number of bandwidth units (BU's) as shown in Fig. 1(a). Here the concept of a bandwidth unit is generic; it may be a time slot in TDMA (Time-Division Multiple Access), a frequency carrier in FDMA (Frequency-Division Multiple Access), or a spreading code in CDMA (Code-Division Multiple Access) (Stuber, 2001, chap. 13). There are two kinds of traffic arrivals in each cell: new connection requests (i.e., sessions originated within the cell and initially requesting new connections), and handoff requests (i.e., sessions migrating from neighboring cells into that cell). Based on the availability of the system resources and the allocation criteria, an admission policy decides at each arrival whether to admit or reject the request. 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. However, the extension of the framework presented in this paper to the general case is straightforward. In each cell, there are two arrival streams: one represents the aggregate handoff traffic migrating from neighboring cells into the current cell with a mean arrival rate λh, and the other represents the new connection requests with a mean arrival rate λn. An ongoing session may leave the system as a result of successful termination or a handoff out of the cell into a neighboring cell. Since the system has a finite capacity, a new connection request or a handoff request may be denied access to the system resources. Assume Bn and Bh refer to the blocking probability of new connection requests and the handoff failure probability, respectively. Our objective is to find a control policy that minimizes a weighted linear function of Bn and Bh as defined by

![Fig. 1. System view and state transition for traffic model in one cell: λ = λh + λn, for i ∈ (0, 1, 2, ..., C − 1) and μ = μh + μn.](image-url)
The transition rate diagram for any allocation policy is shown in Fig. 1(b). The traffic model for a particular cell with a fixed allocation policy is as follows. 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 any allocation policy is shown in Fig. 1(b), where \( \beta_{ij} \) and \( \beta_{in} \) refer to the probability of admitting a new connection and handoff request in state \( i \), respectively. Let \( \lambda_{ij} \) be the transition rate from state \( i \) to \( j \). The steady state probabilities can be expressed as,

\[
P_i = \prod_{j=1}^{C} \frac{\lambda_{ji} \beta_{ij}}{\lambda_{ij} \beta_{ij} + \lambda_{in} \beta_{in}},
\]

where the normalization constant \( P_0 \), the probability that no channel is occupied, can be found from,

\[
P_0 = \left[ 1 + \sum_{i=1}^{C} \prod_{j=1}^{C} \frac{\lambda_{ji} \beta_{ij}}{\lambda_{ij} \beta_{ij} + \lambda_{in} \beta_{in}} \right]^{-1}.
\]

The blocking probability, \( B_n \), and the handoff failure probability, \( B_h \), are given by,

\[
B_n = \sum_{i=0}^{C} (1 - \beta_{in}) P_i + P_C,
\]

\[
B_h = \sum_{i=0}^{C} (1 - \beta_{in}) P_i + P_C.
\]

The aim of any allocation control strategy is to determine the parameters \( \beta_{in} \) and \( \beta_{in} \) for all the states. For complete sharing \( \beta_{in} = \beta_{in} = 1 \) for \( i = 0, 1, 2, \ldots, C - 1 \) and zeros otherwise. Hence, the blocking and the handoff failure probabilities are the same (i.e., \( B_n = B_h = P_C \)). On the other hand, for the guard channel approach the admission policies are different: \( \beta_{in} = 1 \) for \( i = 0, 1, 2, \ldots, C - 1 \) and zero otherwise, whereas \( \beta_{in} = 1 \) for \( i = 0, 1, 2, \ldots, G - 1 \) and zero otherwise where \( G \) is a guard threshold value. In this case, the blocking probability and handoff failure probability are given by,

\[
B_n = \sum_{i=G}^{C} P_i \quad \text{and} \quad B_h = P_C.
\]

4. Problem formulation as average cost SMDP

The problem can be formulated as an infinite-horizon finite-state semi-Markov decision process (SMDP) under the average cost criterion. Since it is always optimal to accept a handoff request as long as there is a free channel, in the following we only find an admission policy for new connection requests. The decision maker learns an admission policy from direct interaction with the network. The primary components of an average cost SMDP are defined as follows. The decision epochs occur at discrete times when new connection requests arrive. 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 inter-arrival times. The system state is defined as the number of busy channels immediately prior to a new connection request arrival, i.e., before making a decision. The system state may change between two decision epochs due to handoff arrival or session termination (whether successfully terminates or hands-over outside the cell). However, only the system state immediately prior to a new connection request is 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 \( A = \{0 = \text{reject}, 1 = \text{admit}\} \) for \( s \in \{0, 1, 2, \ldots, C - 1\} \) and \( A = \{0 = \text{reject}\} \) for \( s \in \{C\} \). The flexibility of the proposed approach allows the system state and actions to be defined as vectors in general to handle multi-dimensional models where other factors can be considered. For example, when the channel allocation policy is not static, the channel status in neighboring cells can be included. Also when there are diverse traffic classes, the number of channels allocated for each class can be included.

A deterministic stationary policy is a mapping from states to actions \( \pi : S \rightarrow A \). Starting from initial state \( s_0 = i \) and implementing policy \( \pi \), the system state at \( n \)th decision epoch, \( s_n \), evolves as a semi-Markov chain \( \{s_n : n \geq 0, s_0 = i\} \) with a state transition probability \( P(s_{n+1} = j | s_n = i, a = \pi) = P_{ij}^\pi \), that is, the probability that the system state at the next decision epoch is \( j \) given that the current state is \( i \) and action \( \pi \) is chosen. At each decision epoch the controller incurs a random stage cost which is the sum of a lump cost as well as an accumulated cost. The lump cost is received immediately and depends on the decision whether to admit or reject the new connection request. The accumulated cost depends on the number of rejected handoff requests which is not known until the next decision epoch. The sequence of incurred costs is a stochastic process and depends on the adopted policy. Assume the system starts at time \( t_0 \) in state \( s_0 = i \) and follows some policy \( \pi \), then the long-term average cost is defined as

\[
g^\pi(i) = \lim_{n \rightarrow \infty} E \left[ \sum_{k=1}^{n} c_k \right] / E \left[ \sum_{k=1}^{n} t_k \right],
\]

where \( c_k \) is the immediate cost incurred over the time interval \( [t_{k-1}, t_k) \) under policy \( \pi \) and \( t_k = t_k - t_{k-1} \). For ergodic Markov decision process, the average cost exists and is independent of the starting state (Hong & Rappaport, 1986), i.e.,

\[
g^\pi(x) = g^\pi(y) \quad \forall x, y \in S \quad \text{and} \quad \forall \pi \in \Pi.
\]

Here \( \Pi \) is a set of all feasible policies and \( g^\pi \) is the long-term average cost under a particular policy \( \pi \). Let \( p^\pi_i \) be the steady-state probability that the system occupies state \( i \) under policy \( \pi \), then the long-term average cost is given by,
The corresponding optimal policy is applying action \( s^* \) in state \( i \) and \( c^* \) is the expected immediate cost incurred when applying action \( a \) in state \( i \). The controller objective is to determine a stationary admission policy in order to minimize the long-term average cost. Formally, we are seeking a policy \( \pi^* \) with a corresponding long-run average cost which is optimal, i.e.,

\[
g^* = \min_{\pi \in \Pi} g^\pi.
\]

For finite state and action spaces, an optimal policy has the property that, no matter what are the initial state and decision, the remaining decisions must form an optimal policy with regard to the resulting state from the first transition. The Bellman optimality equations for the average cost ergodic Markov decision process have the recurrence form

\[
h^*_s = \min_{a \in A_s} \left\{ c^*_s - g^* \tau^i + \sum_{y \in S} P_{sy} h^*_y \right\} \quad \forall x \in S.
\]

where \( h^*_s \) is an optimal state dependent value function \( h^*: S \rightarrow \mathbb{R} \). The corresponding optimal policy is

\[
\pi^*_s = \arg \min_{a \in A_s} \left\{ c^*_s - g^* \tau^i + \sum_{y \in S} P_{sy} h^*_y \right\} \quad \forall x \in S.
\]

Now if a perfect model (the transition probabilities, the sojourn times, and the cost functions) can be analytically derived; then, the solution of the set of Eqs. (12) can be obtained through dynamic programming techniques. Once the optimal state values are obtained, the corresponding policy is attained from (13). However, for complex dynamic cellular networks supporting multimedia traffic with diverse requirements and characteristics, analytical models can be very difficult to derive. Also, the computational complexity precludes the classical dynamic programming optimization techniques to scale well for large state spaces. In the next section, we present a new paradigm based on the reinforcement learning methodology for approximating the optimal solution of (12) online.

5. Proposed model-based reinforcement learning schemes

The model-based learning architecture, as illustrated in Fig. 2, has three main components: controlled system, policy learner, and model estimator. The controlled system could be a real-world or simulated system. The policy learner has two elements: value functions and policy finder. The value functions are real-valued mapping of the system states or state-action pairs. The control policy is a mapping of the system states to the corresponding actions.

5.1. 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 \( N_{xy}(k) \) be the number of times the system state changes from \( x \) to \( y \) under action \( a \) before the \( k \)th decision epoch; \( N_{xy}^i(k) \) be the number of times of executing action \( a \) in state \( x \) before the \( k \)th decision epoch. Then, the controlled state transition probabilities at the \( k \)th decision epoch are estimated as follows:

\[
P_{xy}(k) = \begin{cases} \frac{N_{xy}(k)}{N_{xy}^{|a|}(k)} & \text{if } x = x_{k-1}, y = x_k \land a = a_{k-1} \\ P_{xy}(k-1) & \text{otherwise} \end{cases}
\]

and the average immediate cost is estimated as

\[
c^*_s(k) = \begin{cases} c^*_s(k-1) + \frac{c_{s}(q_{s}-x_{k-1})}{N_{xy}(k)} & \text{if } x = x_{k-1} \land a = a_{k-1} \\ c^*_s(k-1) & \text{otherwise} \end{cases}
\]

Similarly, the average sojourn time is given by

\[
\tau^i_s(k) = \begin{cases} \tau^i_s(k-1) + \frac{\tau_{x_{k-1}}}{N_{xy}(k)} & \text{if } x = x_{k-1} \land a = a_{k-1} \\ \tau^i_s(k-1) & \text{otherwise} \end{cases}
\]

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

5.2. Average cost estimation

Since the controller has no information about the long-term average cost \( g^* \), it can be estimated online at each decision epoch using the following update rule

\[
\rho_k = (1 - \beta) \rho_{k-1} + \beta \left( c_k + h_k - h_{k-1} \right),
\]

where \( \beta \in [0, 1] \) is a step size parameter and \( h_i \) is an estimate of \( h^*_i \). Another approach to estimate \( g^* \), using accumulated costs and times only when selecting greedy actions, is

\[
\rho_k = \sum_{i=1}^k c_{i} l_{i-1} / \sum_{i=1}^k \tau_{i} l_{i-1},
\]

where \( l_{i-1} \), an indicator function, equals one if a greedy action is applied at the \( t-1 \)th decision epoch and zero otherwise.

In the following subsection, we develop a family of adaptive channel allocation control strategies based on the estimated model.

5.3. Learning the optimal value functions

Given an approximate system model as in (14)–(16), in this subsection we develop three fundamental schemes for learning the optimal value functions and hence the optimal policy. The first two learning schemes update the value functions using the principle of certainty equivalence (Ramjee et al., 1997) while the third one uses a gradient-like scheme to incrementally update the action value functions. Other variations are also possible, e.g., we can combine different schemes or run different techniques at different phases of the system lifetime.

5.3.1. Model-based average-cost reinforcement learning (mARL) scheme

This method is simple and based on the principle of certainty equivalence. The policy learner uses the estimated model as the
true model and updates the $h$-values in (12) at each decision epoch as follows:

$$h_x(k) = \min_{a \in A_x} \left\{ c^a_x + \rho_x \tau^a_x + \sum_{y \in S} p^a_{xy} h_y(k - 1) \right\}. \tag{19}$$

Depending on the time available to the controller until the next decision, the update rule in (19) can be applied only to the currently observed state, similar to asynchronous DP approach, or it may be applied to a subset of states using prioritized sweeping as in Moore and Atkeson (1993).

5.3.2. Model-based average-cost Q-learning (mAQ) scheme

Another version similar to mARL is to learn the action values instead of the state values. This scheme is based on the Q-learning algorithm which has been first devised by Watkins (1989) for discrete-time Markov decision processes (MDP) under the discounted total framework. Schwartz (1993) has proposed a counterpart learning approach for the undiscounted optimization problem called R-learning. Singh (1994) has derived a new version of R-learning for average-payoff MDP. In contrast to the average cost Q-learning (Singh, 1994), the new scheme is for the continuous time Markov decision process and utilizes the estimated model to facilitate the policy learning on-line. Like the average cost Q-learning, the optimal Q-functions are defined for each state-action pair as

$$Q_x^a = c^a_x - g^x \tau^a_x + \sum_{y \in S} p^a_{xy} h^y(y) \quad \forall x \in S \text{ and } a \in A_x. \tag{20}$$

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

$$h_x^* = \min_{a \in A_x} Q_x^a \quad \forall x \in S. \tag{21}$$

Hence, Eq. (20) becomes

$$Q_x^a = c^a_x - g^x \tau^a_x + \sum_{y \in S} p^a_{xy} \min_{b \in A_y} \left\{ Q_y^b \right\} \quad \forall x \in S \text{ and } a \in A_x. \tag{22}$$

Again based on the principle of certainty equivalence, the controller uses the approximate model to update the Q-values at each decision epoch as follows:

$$Q_x^a(k) = c^a_x(k) - \rho_x \tau^a_x(k) + \sum_{y \in S} p^a_{xy}(k) \min_{b \in A_y} \left\{ Q_y^b(k - 1) \right\}. \tag{23}$$

Like the mARL scheme the updates (23) can be applied to one or more state-action pairs. Learning the Q-values will reduce the time required to make a decision upon the arrival of a new connection request.

5.3.3. Incremental mAQ (imAQ) scheme

Integrating the idea of temporal difference used in the model-free learning and the model-based learning approach can lead to further improvements in the system performance. We use the difference between the approximate value given by (23) and the previous value to update the Q-values incrementally similar to the stochastic gradient approach. The error between the new value and the old value is computed using

$$\Delta Q_x = c^a_x(k) - \rho_x \tau^a_x(k) + \sum_{y \in S} p^a_{xy}(k) \min_{b \in A_y} \left\{ Q_y^b(k - 1) \right\} - Q_x^a(k - 1). \tag{24}$$

Hence, the Q-values will be updated as follows

$$Q_x^a(k) = Q_x^a(k - 1) + \alpha \Delta Q_x. \tag{25}$$

where $\alpha \in [0, 1]$ is a learning rate or a step-size parameter. The setting for $\alpha$ can be fixed or gradually decaying over time. Again there is no restriction to apply rules (24) and (25) to one or more state-action pairs 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. One direct way is to use the generated sample data to update the value functions (using temporal difference approach) and to update the learned model. Then, use the learned model to make predications and generate simulated sample data which can be used to update the value functions (Sutton & Barto, 1998). Another approach is based on the observation that the error in the approximate model is large in the early stages; therefore, the model-free schemes are more efficient 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 approaches. But determining the switching point is not a trivial task.

5.4. Reactive policy finder

When a new connection request arrives, the controller observes the state of the network, i.e., the number of occupied channels, and determines whether to admit or reject based on the state and state-action value functions. Various approaches can be used to select an action using the learned state or 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. For example, if the system state at the kth decision is $x_k$ when a new connection request arrives, the decision is made according to

$$\pi_{x_k} = \arg \min_{a \in A_{x_k}} \left\{ Q^a_{x_k}(k) \right\}. \tag{26}$$

This channel allocation control scheme is summarized in the pseudo code in Fig. 3. Since the learned model and policy are based on sampling, the accuracy depends on the generated trajectories which in turn depend on the exploited policy. This means that the learning agent needs to explore while learning, i.e., to try apparently sub-optimal actions some of the time. However, inefficient exploration may cause the algorithm to be trapped into sub-optimal solutions. To allow the learner to maintain a balance between exploiting the past experience and exploring for new information, more complicated selection mechanisms are needed. One approach is called $\epsilon$-greedy in which the greedy action is selected with high probability, $1 - \epsilon$, and with small probability, $\epsilon$, uniformly select among actions.

---

Please cite this article in press as: El-Alfy, E.-S. M., & Yao, Y.-D. Comparing a class of dynamic model-based reinforcement learning schemes for handoff prioritization in mobile communication networks. Expert Systems with Applications (2011), doi:10.1016/j.eswa.2011.01.082
One variation of this approach is to start with a large value of $e$ and decrease it over time as more information is gained. Other approaches for balancing the exploration and exploitation are presented in (Thrun, 1992). For example the action selection can be made according to the Boltzmann’s distribution where the exploration rate is controlled through the temperature parameter.

6. Simulation and numerical results

In this section, we conduct intensive simulation runs to empirically evaluate and compare the performance of the learning algorithms with the complete sharing (CS) and the optimal guard channel 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 connection and handoff requests according to mutually independent Poisson processes. In the first experiment, the mean arrival rates are set to $\lambda_0 = 5$ requests/min, and $\lambda_h = 3$ requests/min respectively. Each session 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 $C_0 = 18$ and the corresponding blocking probabilities $B_h = 0.010128$ and $B_n = 0.150796$. We ran the simulator for 10 runs and the average values are depicted in Figs. 4–6. The average cost incurred by each policy is shown in Fig. 4 whereas Figs. 5 and 6 show the corresponding blocking probability and handoff failure probability. The value functions are initialized to zeros and the step-size parameter $\alpha$ is set to 0.01. The simulation results show that the learning approaches were capable of self-adjusting and prioritizing the handoff. However, the performance of the incremental model-based Q-learning (imAQ) is better than mARL and mAQ. Furthermore, the imAQ scheme outperforms the complete sharing policy and is very close to the optimal guard policy. To test the performance of the learning approaches compared to other policies for different traffic conditions (e.g., when the handoff rate changes), we ran 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, blocking probability and handoff failure probability are plotted versus the handoff rate in Figs. 7–9. Again as depicted in Fig. 7 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.

7. Conclusions

In this paper, we have addressed the handoff prioritization problem in mobile communication networks. The problem has been formulated as an average cost continuous-time Markov decision problem. Three model-based reinforcement learning schemes have been developed and compared. Simulation results show that
for the considered traffic conditions, the proposed schemes can autonomously adjust the allocation policy to the traffic conditions. Also, imAQ scheme outperforms the complete sharing policy and the other learning approaches and has a comparable performance to the optimal guard channel approach. Another observation is that when the handoff rate is low (most ongoing sessions terminate in the originating cell before requesting handoff), all policies have almost the same performance as complete sharing (because only very few channels need to be reserved). However, as the handoff rate increases, the guard channel and imAQ become more effective. Although in this paper we assumed that the value functions are particle swarm optimization and fuzzy logic systems. Expert Systems with Applications, 35(3), 1246–1251.


Fig. 8. The effect of handoff rate on the performance of imAQ policy against other policies: blocking probability for new connection requests.

Fig. 9. The effect of handoff rate on the performance of imAQ policy against other policies: handoff failure probability.

Acknowledgment

The first author would like to thank King Fahd University of Petroleum and Minerals (KFUPM), Dhahran, Saudi Arabia for continuous support during this work.

References


Please cite this article in press as: El-Alfy, E.-S. M., & Yao, Y.-D. Comparing a class of dynamic model-based reinforcement learning schemes for handoff prioritization in mobile communication networks. Expert Systems with Applications (2011), doi:10.1016/j.eswa.2011.01.082


