Call Admission Control for Reducing Dropped Calls in CDMA Cellular Systems

Yue Ma [a], James J. Han [a] and Kishor S. Trivedi [b]

[a] *Global Software Group, Motorola, Inc., 50 NW Point Blvd., 2FL, Elk Grove Village, IL 60007-1032, USA*

[b] *Center for Advanced Computing and Communication (CACC), Department of Electrical and Computer Engineering, Duke University, Durham, NC 27708, USA*

Abstract

Call admission control algorithms that reduce dropped calls in CDMA cellular systems are discussed in this paper. The capacity of a CDMA system is confined by interference of users from both inside and outside of the target cell. Earlier algorithms for call admission control is based on the effective traffic load for the target cell if one call is accepted. These algorithms ignore the interference effect of the to-be-accepted call on the neighboring cells. In our algorithms, the call admission decision is based on the effective traffic loads for both the target cell and the neighboring cells. In addition, to prioritize handoff calls, we also introduce the idea of *soft guard channel*, which reserves some traffic load exclusively for handoff calls. Stochastic reward net (SRN) models are constructed to compare the performance of the algorithms. The numerical results show that our algorithms can significantly reduce the dropped calls with a price of increasing the blocked calls. To show the potential gain due to our algorithms, we introduce two new metrics: *the increased blocking ratio* for our algorithms and *the increased dropping ratio* for the conventional algorithms. From the numerical results, it is shown that our algorithms can reduce the dropped calls significantly while the blocked calls are increased at a relatively small rate under both homogeneous and hot spot traffic loads.

Key words: call admission control, CDMA, fixed-point iteration, interference, soft guard channel, Markov reward models, stochastic reward net (SRN).

Email addresses: `yue.ma@motorola.com` (Yue Ma), `cjh048@email.mot.com` (James J. Han), `kst@ee.duke.edu` (Kishor S. Trivedi).

1 Introduction

Since the early 1990's, personal communications services (PCS) have been growing at an explosive rate. To meet the continuing growth of the customer demand, the limited spectrum needs to be used as efficiently as possible. Spread spectrum code-division multiple access (CDMA) scheme provides high spectrum efficiency [26]. In a CDMA system, all the users share the same radio frequency at the same time, the system capacity is strictly interference limited. Unlike a frequency-division multiple access (FDMA)/time-division multiple access (TDMA) system where capacity is fixed due to the frequency/time allocation, the capacity of a CDMA system is soft. As the number of users sharing the same radio frequency is increased, the signal-to-interference ratio (SIR) generally degrades. Therefore, to some extent, the capacity of a CDMA cellular system depends on how sensitive the subscriber is to the background noise.

Call admission control (CAC) algorithms for reducing dropped calls in CDMA systems are discussed in this paper. The capacity of a CDMA system is confined by interference of users from both inside and outside of the target cell. The earlier CAC algorithms [11,15,22] are based on the effective traffic load for the target cell if the call is accepted. However, these methods ignore the interference effect of the to-be-accepted calls from the neighboring cells. The following example shows the potential negative effect of this ignorance.

Suppose that cell A and cell B are in a common neighborhood. The traffic load in cell B has already reached its maximum, while the traffic load in cell A is quite light. All the new call arrivals in cell B will be denied because it has already reached its saturation point. According to the earlier method (we refer to it as the Self-Decision algorithm), any new call arrivals will be accepted in cell A because it has not reached its maximum load yet. Since cell B has already reached its maximum load, accepting a new call in cell A could possibly deteriorate the quality of the traffic channels in cell B beyond tolerance and consequently causing dropped calls or outage. Hence, to maintain the overall Quality of Service (QoS) for the whole system, it is necessary to consider the interference impact on the neighboring cells when accepting a call in a cell.

In this paper, we propose a method (called the Looking Around algorithm) for call admission control in CDMA systems. In this method, the call admission decision is based on the effective traffic loads for both the target cell and the neighboring cells. From the numerical results, it is shown that the LA algorithm can significantly reduce the probability for dropped calls. However, we do need to pay a price: an increased blocking probability. This is well justified because when the system is heavily loaded, the acceptance of a new call (NC) in one cell would cause possible dropping or outage in adjacent cells.

To analyze the tradeoff between the decreased dropping probabilities and the increased blocking probabilities, we introduce two new metrics, namely, *the decreased dropping ratio* and *the increased blocking ratio*. From the numerical results, we can see that the gain for the decreased dropping probabilities is worthwhile. From the numerical results, it is shown that our algorithms can reduce the dropped calls significantly while the blocked calls are increased at a relatively small rate, under both the homogeneous and non-homogeneous traffic.

Another contribution of this paper is that we introduce the idea of *soft guard channel* to prioritize soft handoff calls (SHC). In FDMA/TDMA systems, physical *hard* guard channels in forms of frequencies/time slots are reserved exclusively for handoff calls. In CDMA cellular systems, the capacity is interference limited. There are no physical resources which can be used as guard channels. Instead, a certain amount (either integral or fractional) of traffic load, denoted as *soft guard channels*, is reserved exclusively for soft handoff calls.

This paper is organized as follows. In Section 2, some of the related features of a CDMA cellular system are described. In Section 3, we present our CAC algorithms for reducing dropped calls. To obtain numerical results for the CAC algorithms, stochastic reward nets (SRNs) are developed in Section 4. A brief introduction of SRN is first given at the beginning of that section. In Section 5, the performance of the LA and the SD algorithms are compared. Finally, conclusions are made in Section 6.

2 System Description

2.1 Soft Handoff

In cellular networks, different multiple access schemes are used to provide resources (channels) for setting up/maintaining calls. In FDMA, each channel can be regarded as a different frequency. In TDMA, the calls are served with different time slots. In CDMA, the calls are served with different pseudonoise (PN) sequences. When a mobile station (MS) travels across the cell boundaries, the channel in the old serving cell is released, and an idle channel is required in the target cell, which would be the new serving cell. This process is called handoff. In a conventional cellular network, the MS breaks the communication link with the old base station (BS) before establishing a new communication link with the new BS. This kind of handoff process (break before make) is generally experienced with a brief interception of communication and is commonly referred as *hard handoff*.

In a CDMA system, a so called *soft handoff* mechanism is utilized. Each MS measures the signal strength from its surrounding BSs. The PN offsets of the BSs serve as the identification numbers for the BSs. When an adjacent base station's pilot strength is strong enough to establish a communication link, the base station's PN offset is stored in the mobile station's Active Set. With lower signal thresholds, the MS also maintains a Candidate Set and a Neighborhood Set. When an MS is undergoing a soft handoff, more than one pilot is stored in the Active Set. The MS communicates with two or more BSs until the communication link between the mobile and one base station is firmly established. The "Ping-Pong" effect (constant handing back and forth between base stations at the border), a common phenomena in hard handoff, is avoided under this "make before break" strategy. Thus, the signal quality and service reliability is substantially improved through soft handoff.

2.2 CDMA Cellular System Model

Assume that a target cell in a wireless network can be partitioned into two zones: core zone (CZ) and the soft handoff zone (SHZ). With respect to the base station in the target cell, the area immediately adjacent to the SHZ is called the neighborhood zone (NZ). Figure 1 shows a concentric geometry used for the purpose of illustrating the relationship among the three zones. In [14], similar concentric geometry approaches are used for analyzing the performance of cellular networks.

When an MS is in soft handoff phase, we assume that at most two base stations are in diversity reception. For the purpose of discussion, new call interarrival times are assumed to be independent and exponentially distributed [13], with rate λ_n . A new call may arrive in either a core zone or a soft handoff zone.

Assume that new calls are uniformly distributed in the target cell, the new call arrival rates for the two zones are:

$$\lambda_n^c = a\lambda_n, \quad \lambda_n^s = (1-a)\lambda_n$$

where λ_n^c or λ_n^s is respectively the new call arrival rate in CZ or SHZ and $a = (area\ of\ CZ)/(area\ of\ target\ cell)$. Suppose that the mean target cell dwelling time is μ_d, then the average time that an MS spends in CZ and SHZ is respectively $a\mu_d$ and $(1-a)\mu_d$. For reasons of tractability [7], the call holding times are assumed to be exponentially distributed.

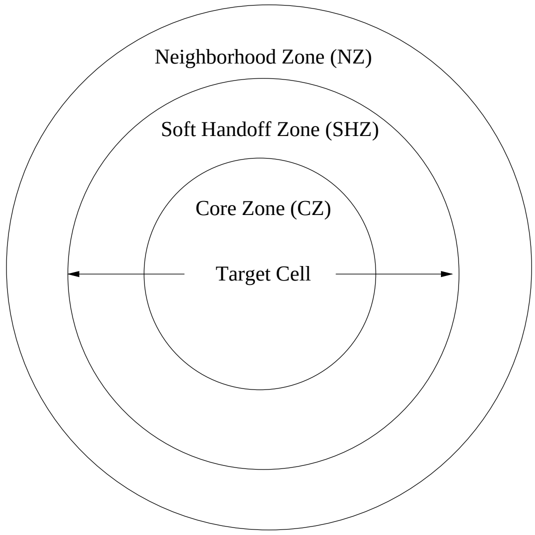

Fig. 1. A concentric geometry for a target cell and its neighborhood area.

3 Call Admission Control (CAC) Algorithms through *Looking Around*

When a NC/SHC arrives, one method for call admission is based on local information without 'consulting' the neighborhood. We refer to this as the SD algorithm because the call admission is based on self-decision. We enhance the SD algorithm by considering the possible interference from the neighborhood cells. The enhanced algorithm is denoted as the LA algorithm because the call admission is made after 'looking around' at the neighborhood cells.

3.1 CAC Algorithm for New Calls

Figure 2 shows the CAC algorithm for the new calls. When a new call arrives, the base station at the target cell first checks if it is in the CZ or SHZ. This can be achieved through two ways. One way is to obtain the information by the signal strength. The other is to check the number of pilots in the Active Set of the new call. The effective load for the target cell (ELT) is then calculated through the following equation:

$$ELT = k_c + w_s \cdot k_s + w_n \cdot k_n$$

where k_c (k_s) is the number of calls in the CZ (SHZ) before a possible admission of an NC; k_n is the number of calls in the NZ; w_s and w_n are the weights and $0 < w_n \leq w_s \leq 1$. If the target cell is already saturated, the new call is blocked. Otherwise, if the NC is in the SHZ, the effective load for the other cell (ELS2) that covers the SHZ is also calculated. Again, the threshold is

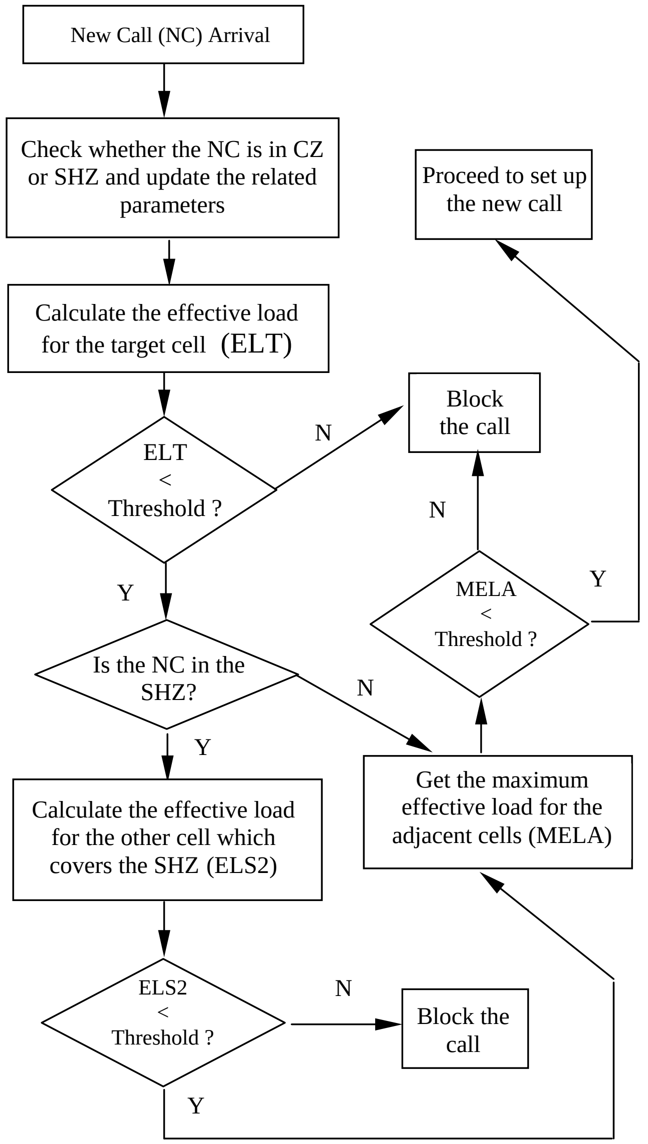

Fig. 2. CAC algorithm for the new call arrivals in the target cell.

compared with ELS2, if no saturation point is reached for the other cell, the algorithm continues with the final check for the neighborhood cells. The call is accepted if the maximum effective load for the adjacent cells (MELA) is below the threshold.

Notice that in [22], if a new call is denied in the SHZ of one cell, it is allowed to be set up through another cell whose SHZ also covers the call. This algorithm would have the risk of dropping some ongoing calls in the already saturated cell.

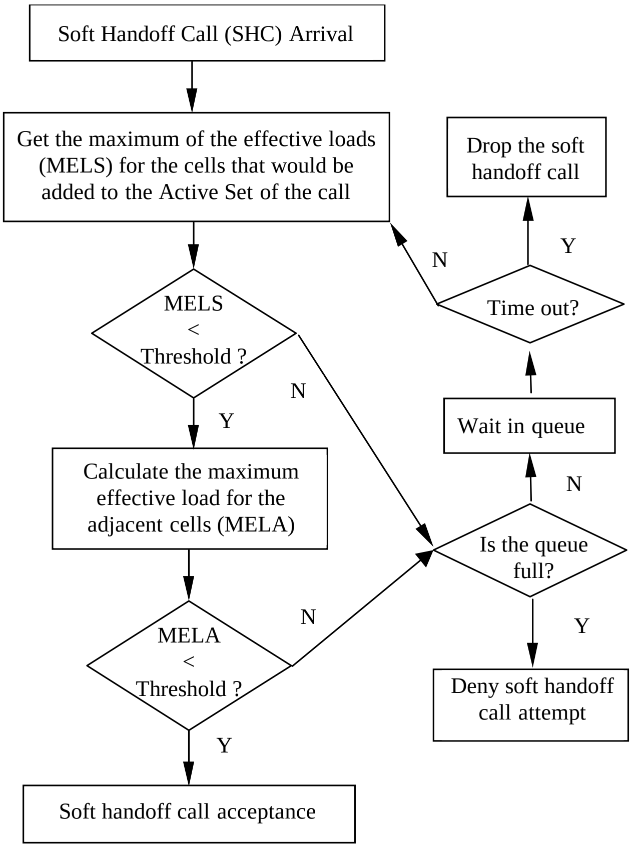

Fig. 3. CAC algorithm for soft handoff calls in the target cell.

3.2 CAC Algorithm for Soft Handoff Calls

The CAC algorithm for soft handoff calls is shown through Figure 3. When a soft handoff call is required, the effective loads for the cells that cover the SHZ are calculated. The largest value from these (denoted as MELS) is compared with the predetermined threshold. If there is still room for the cells, the algorithm continues with calculating the maximum effective load for the adjacent cells. If again the MELA is under the threshold, the soft handoff call is accepted and diversity combination is applied for receiving the call. If either MELS or MELA is above the threshold, the SHC requirement would be put in a queue if the queue is not full. Otherwise, the SHC is denied. While waiting in the queue, the SHCs are periodically checked on whether the time out event has occurred or not. If a time out event has occurred before the SHC can obtain access to the target cell, it is dropped.

3.3 Soft Guard Channel

The dropping of a handoff call (which is an ongoing call) is considered more disturbing than the blocking of a new call. Prioritizing handoff calls [23,24] is an important way to reduce handoff failures. In our CAC algorithm, we introduce the idea of *soft guard channel* to prioritize handoff calls. In addition, queuing [21] is also used to enhance the handoff success probability. Unlike FDMA or TDMA systems, which use frequency or time slot as resources for 'hard' guard channels, the resources in CDMA cellular systems are interference limited. There are no physical resources in CDMA systems that can be used for guard channels. To prioritize SHCs, a certain amount of traffic capacity is reserved exclusively for SHCs. In the remainder of this paper, we refer to the reserved traffic capacity as the *soft guard channels* (SGC). Assume that the threshold in Figure 2 is denoted as TN and the threshold in Figure 3 is denoted as TS. Using soft guard channels to prioritize SHC, TS is always greater than or equal to TN. Denoting the numerical value of SGC as g, we have $g = TS - TN$. The value of g can be either integral or fractional depending on the traffic volume of the environment.

4 Stochastic Reward Net Models for the CAC Algorithms

4.1 Introduction to Stochastic Reward Net (SRN)

In this paper, we use Markov reward model as the modeling paradigm. In order to automatically generate and solve Markov reward models, we use the framework of stochastic reward net (SRN). SRN has the advantage of specifying a real-world model in a compact and intuitive way. Since the early 1990's, SRN has been used as a powerful modeling tool in performance, availability and reliability analysis in communications systems [9,10,12,17,19,27]. In this section, we give a brief introduction to SRN.

Stochastic reward net (SRN) [4] is an extension of Petri net (PN) [1,3], which is a high level description language for formally specifying complex systems. A PN is a bipartite directed graph with two types of nodes: *places* and *transitions*. Each place may contain an arbitrary (natural) number of *tokens*. For a graphical presentation, places are depicted as circles, transitions are represented by bars and tokens are represented by dots or integers in the places. Each transition may have zero or more *input arcs*, coming from its input places; and zero or more *output arcs*, going to its output places. A transition is *enabled* if all of its input places have at least as many tokens as required by the multiplicities of the corresponding input arcs. When enabled, a transition can *fire* and will remove from each input place and add to each output place the number of tokens corresponding to the multiplicities of the input/output arcs. A *marking* depicts the *state* of a PN which is characterized by the assignment of tokens in all the places. With respect to a given *initial* marking, the *reachability set* is defined as the set of all markings reachable through any possible firing sequences of transitions, starting from the initial marking.

Generalized stochastic Petri nets (GSPNs) [1] extend the PNs by assigning a *firing time* to each transition. Transitions with exponentially distributed firing times are called *timed* transitions while the transitions with zero firing times are called *immediate* transitions. A marking in a GSPN is called *vanishing* if at least one immediate transition is enabled; otherwise it is called a *tangible* marking. For a given GSPN, an *extended reachability graph* ($\mathcal{ERG}$) is generated with the markings of the reachability set as the nodes and some stochastic information attached to the arcs, thus connecting the markings to each other. Under the condition that only a finite number of transitions can fire in finite time with non-zero probability, it can be shown that a given $\mathcal{ERG}$ can be reduced to a homogeneous continuous time Markov chain (CTMC) [1,3].

In order to make more compact models of complex systems, several extensions are made to GSPN, leading to the SRN. One of the most important features of SRN is its ability to allow extensive marking dependency. In an SRN, each tangible marking can be assigned with one or more *reward rate(s)*. Parameters such as the firing rate of the timed transitions, the multiplicities of input/output arcs and the reward rate in a marking can be specified as functions of the number of tokens in any place in the SRN. Another important characteristic of SRN is the ability to express complex enabling/disabling conditions through *guard* functions. This can greatly simplify the graphical representations of complex systems. For an SRN, all the output measures are expressed in terms of the expected values of the reward rate functions. To get the performance and reliability/availability measures of a system, appropriate reward rates are assigned to its SRN. As SRN is automatically transformed into a Markov reward model (MRM) [4,25], steady state and/or transient analysis of the MRM produces the required measures of the original SRN. In this paper, we use the tool Stochastic Petri Net Package (SPNP) [3,5] to specify and solve the SRN models.

4.2 Model Description

In this section, we construct SRN models in accordance with the CAC algorithms described in Section 3. Because of the interaction among the cells, an SRN model for even a cluster of cells (Figure 4) would instigate a huge state space for the underlying Markov reward process. Depending on the nature

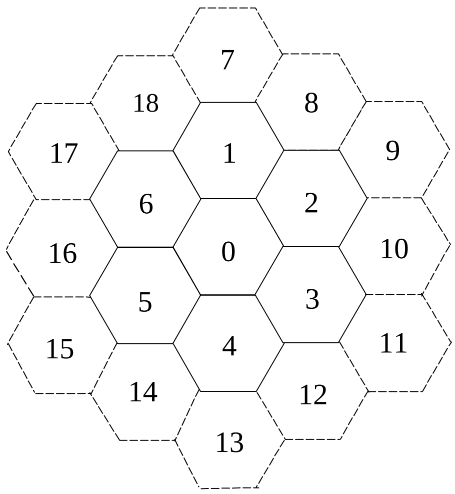

Fig. 4. A cluster of cells for analytical study.

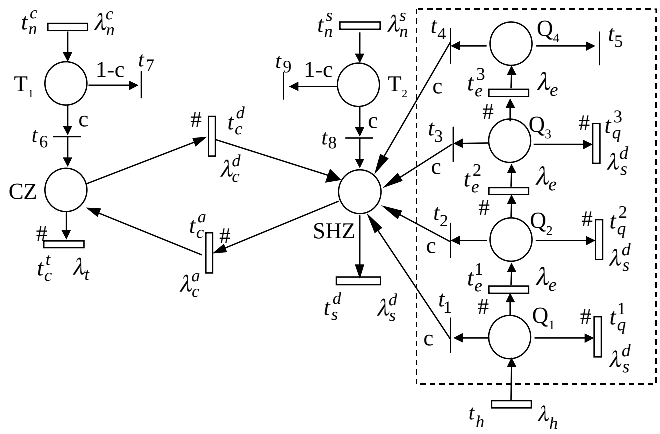

Fig. 5. An SRN model for a generic cell in a CDMA cellular network with LA algorithms for CAC.

of the system to be modeled, different approaches can be applied to reduce the underlying state space for SRN models. Examples for these approaches are: state truncation [20]; hierarchical approach [16] and decomposition approach [6]. In this paper, our SRN models are based on a decomposition approach. Figure 4 shows a cluster of seven cells (in solid lines) together with their outside neighborhood cells shown with dashed lines. We will concentrate on the center cell, C_0. The LA algorithm for call admission control in a generic cell can be represented by an SRN model shown in Figure 5.

In Figure 5, places CZ and SHZ respectively represent the core zone and the soft handoff zone for a generic cell. New call arrivals for the CZ and SHZ are represented by the transitions t_n^c and t_n^s, respectively. Whether the current load of the target cell is under the threshold or not is determined by the enabling functions for transitions t_n^c and t_n^s. If the adjacent cells are not overloaded, the new calls are accepted through immediate transitions t_6

and t_8, with probability c. Otherwise, the calls are blocked through immediate transitions t_7 and t_9, with probability $1-c$. Transition t_c^t represents the normal termination of a call while it is in the CZ. A user can move from the CZ to the SHZ and vice versa while talking. These are depicted through transitions t_c^d and t_c^a. With regard to the other cell which covers the SHZ, transition t_c^d also represents a handoff arrival for that cell; while transition t_c^a represents a handoff departure.

While a user is in SHZ, three possibilities exist:

- The user moves to the CZ of the cell being modeled; this is represented by transition t_c^a.
- The user moves to an 'outside' adjacent cell, with rate λ_h^d.
- The call is normally terminated, with rate λ_t. In Figure 5, these last two transitions are combined into one transition t_s^d, with $\lambda_s^d = \lambda_h^d + \lambda_t$.

For a handoff arrival, represented by transition t_h, the call can be accepted immediately through transition t_1 if the cell and its neighborhood are not overloaded. If the effective load is over the threshold and the queue is not full, the SHC can wait in the queue. While waiting in the queue, a call will be dropped if a time out event occurs. The dashed rectangle in Figure 5 describes a queue with time out. The time out event is modeled by a three stage Erlang distribution [2,28]. Transition t_e^i, where $i \in [1,3]$, represents a time delay in the queue. Before a user reaches place Q_4, the user can be accepted (from transitions t_1 to t_4) to SHZ whenever the effective load is under the threshold. In addition, the user may go to another cell or undergo a normal call termination via transitions t_q^1 through t_q^3. Once the user enters place Q_4, the final step in checking the time out event is carried out. If at that moment, the threshold conditions are satisfied, the call is accepted to place SHZ from the immediate transition t_4. Otherwise, the call is dropped through the immediate transition t_5. The loading effects of the adjacent cells are reflected through probability c, which is associated with transitions t_1 through t_4. Because of the nature of the soft handoff call, a denied SHC does not necessarily mean that it is dropped. We will discuss a dropped SHC in a later section.

Transitions t_c^t, t_c^d, t_c^a, t_s^d, t_e^i, and t_q^i have marking dependent firing rates; that is, the actual firing rates for these transitions are proportional to the number of tokens in the respective input places. In SRN, the marking dependent firing rate is represented by the sharp sign, #.

For comparison purposes, an SRN model of the SD algorithm is developed as well. It is shown in Figure 6. The main difference between the LA and the SD algorithms is that SD algorithm does not 'look around' to the neighbors before accepting an NC/SHC. Therefore, compared with Figure 5, immediate

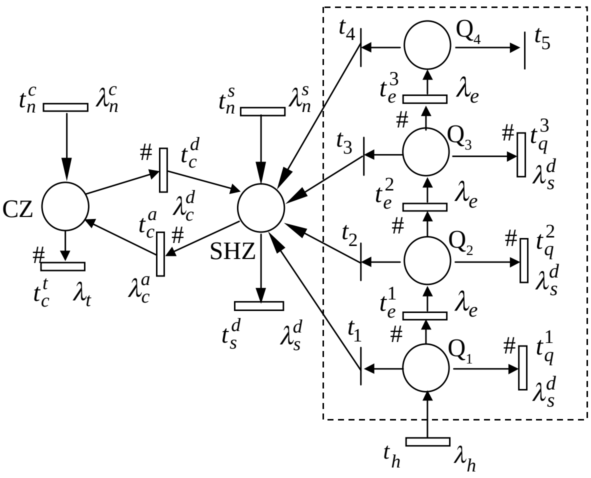

Fig. 6. An SRN model for a generic cell in a CDMA cellular network with SD algorithms for CAC.

transitions t_6 through t_9 and the places T_1 and T_2 are removed from Figure 6. In addition, because the CAC decision in the SD algorithm is based purely on the local information, the probability c is *not* associated with the immediate transitions t_1 through t_4.

4.3 Fixed-point iteration

With different NC arrival rates, the following parameters vary accordingly:

k_n: the total number of MSs in the CZ and the SHZ.
c: the probability that the effective load (EL) of a cell is less than the threshold.
λ_h: handoff arrival rate.

To capture this dynamic behavior, a fixed-point iteration scheme [8,18] is applied to determine the above parameters. The values of these parameters are calculated as following:

$$k_n = \sum_{j \in \Omega} (\#[CZ_j] + \#[SHZ_j]) \cdot \pi_j(k_n, c, \lambda_h), \tag{1}$$

$$c = \sum_{j \in \Omega} r_c^j \cdot \pi_j(k_n, c, \lambda_h), \tag{2}$$

$$\lambda_h = \frac{(\Lambda_s^d + \Lambda_q^1 + \Lambda_q^2 + \Lambda_q^3) \cdot \lambda_h^d}{\lambda_t + \lambda_h^d} + \Lambda_c^d, \tag{3}$$

where Ω is the set of tangible markings in the SRN models and π_j is the steady state probability of marking j. We use Λ to denote the throughput of a transition. For example, Λ_s^d is the throughput of transition t_s^d. In (2), the reward rate assignment r_c^j is of the following form:

$$r_c^j = \begin{cases} 1 \text{ if } (\#[CZ] + w_s \cdot \#[SHZ] + w_n \cdot k_n) \leq T - g, \\ 0 \text{ otherwise.} \end{cases} \quad (4)$$

In (4), # represents the number of tokens in a place, T represents the maximum effective load allowed in a cell, and g is the number of soft guard channels. Figure 7 shows the fixed-point iteration algorithm we use for solving the SRN in Figure 5 under homogeneous traffic. According to Theorem 2 in [18], a fixed point will exist if :

- The iteration function is a weighted sum of state probabilities and the weights are constants;
- The CTMCs underlying the SRNs are irreducible with more than one state.

It is easy to verify that the SRN models developed in this paper satisfy the above two conditions. Therefore, fixed-points exist for (1), (2) and (3). The proofs for the uniqueness of the solutions can be found in [8].

/* initialization, $k_h^{(0)}$, $c^{(0)}$, $\lambda_h^{(0)}$ are initial values. */

1. $k_n^{(1)} \leftarrow k_n^{(0)}$;
2. $c^{(1)} \leftarrow c^{(0)}$;
3. $\lambda_h^{(1)} \leftarrow \lambda_h^{(0)}$;

4. repeat
5.   solve the SRN in Fig. 5;
6.   compute $k_n^{(i+1)}$, $c^{(i+1)}$ and $\lambda_h^{(i+1)}$
    according to (1), (2) and (3), respectively;
7. $\delta_1 \leftarrow |k_n^{(i+1)} - k_n^{(i)}|/k_n^{(i)}$;
8. $\delta_2 \leftarrow |c^{(i+1)} - c^{(i)}|/c^{(i)}$;
9. $\delta_3 \leftarrow |\lambda_h^{(i+1)} - \lambda_h^{(i)}|/\lambda_h^{(i)}$;
10. $\delta \leftarrow \max(\delta_1, \delta_2, \delta_3)$;
11. until $\delta < \epsilon$;

Fig. 7. A fixed-point iteration algorithm for the SRN in Figure 5 under homogeneous traffic.

4.4 Homogeneous traffic

In this section, we assume that all the cells are statistically identical and behave independently. We also assume that the cellular system is undergoing homogeneous traffic. The characteristics of the overall system can be captured by focusing on a single cell.

4.4.1 Enabling function

The NC and SHC arrivals are accepted only if the CAC conditions are satisfied. In SRN, this is achieved through enabling functions and the probabilities associated with the transitions. At transition t_n^c, a new call is accepted under the following conditions:

- $\#[CZ] + w_s \cdot \#[SHZ] + w_n \cdot k_n \leq T - w_s \cdot g - 1$. This condition is associated with the enabling function of t_n^c for both the LA and the SD algorithms.
- Maximum effective load in the adjacent (MELA) cells is less than the threshold. This condition is only required in the LA algorithm. It is carried out through the probability c, which is associated with transitions t_6, t_8, and t_1 through t_4.

The enabling functions for the LA algorithm are summarized in Table 1. The First-Come-First-Served (FCFS) policy is assumed for the queue. The queue length is denoted as le in the table.

4.4.2 Dropping probability

Under homogeneous traffic, we denote the dropping probability for the LA and the SD algorithms as P_{dl}^h and P_{ds}^h, respectively. They are obtained through the following equations:

$$P_{dl}^h = \frac{(1 - c \cdot P_4) \cdot \Lambda_e^3}{\Lambda_h} + \frac{P_q^f \cdot \lambda_h^d/6}{\lambda_t + \lambda_h^d}, \tag{5}$$

$$P_{ds}^h = \frac{(1 - P_4) \cdot \Lambda_e^3}{\Lambda_h} + \frac{P_q^f \cdot \lambda_h^d/6}{\lambda_t + \lambda_h^d} + (1 - c) \cdot P_i, \tag{6}$$

where

$$P_i = \frac{\Lambda_n^c + \Lambda_n^s + \Lambda_h - (1 - P4) \cdot \Lambda_e^3 - \Lambda_q^1 - \Lambda_q^2 - \Lambda_q^3}{\Lambda_n^c + \Lambda_n^s + \Lambda_h}.$$

In the above equations, P_4 is the probability that the enabling function for

Transition	Enabling Function
t_n^c	$\#[CZ] + w_s \cdot \#[SHZ] + w_n \cdot k_n \leq T - w_s \cdot g - 1$
t_n^s	$\#[CZ] + w_s \cdot \#[SHZ] + w_n \cdot k_n \leq T - w_s \cdot (g+1)$
t_1	$\#[CZ] + w_s \cdot \#[SHZ] + w_n \cdot k_n \leq T - w_s$ && $\#(Q_2) == 0$ && $\#(Q_3) == 0$ && $\#(Q_4) == 0$
t_2	$\#[CZ] + w_s \cdot \#[SHZ] + w_n \cdot k_n \leq T - w_s$ && $\#(Q_3) == 0$ && $\#(Q_4) == 0$
t_3	$\#[CZ] + w_s \cdot \#[SHZ] + w_n \cdot k_n \leq T - w_s$ && $\#(Q_4) == 0$
t_4	$\#[CZ] + w_s \cdot \#[SHZ] + w_n \cdot k_n \leq T - w_s$
t_h	$\#(Q_1) + \#(Q_2) + \#(Q_3) < le$

Table 1
Enabling Functions for the LA Algorithms

transition t_4 is enabled, P_q^f is the probability that the queue is full. There are two possibilities that a call can be dropped in the LA algorithm:

(1) The call is dropped after time out, represented by the first term of (5).
(2) The queue is full, and a SHC 'insists' on going to the target cell. This is represented by the second term of (5).

For the SD algorithm, in addition to the above two factors that might cause a call to drop, a call might be dropped if the cell it resides in is full and one of its immediate neighboring cells accepts a call without 'consulting' this target cell. This is represented by the third term of (6).

4.4.3 Blocking probability

Under homogeneous traffic, the blocking probability for the LA and the SD algorithms is denoted as P_{bl}^h and P_{bs}^h, respectively. They are calculated according to the following formulas:

$$P^h_{bl} = (1-a)\cdot P_{bns} + a\cdot P_{bnc} + (1-c)\cdot \frac{\Lambda^c_n + \Lambda^s_n}{\lambda^c_n + \lambda^s_n}, \tag{7}$$

$$P^h_{bs} = (1-a)\cdot P_{bns}\cdot P_{bns} + a\cdot P_{bnc}, \tag{8}$$

where P_{bnc} and P_{bns} respectively represents the probability that the new call is blocked at the CZ or SHZ , if the target cell is overloaded. In the LA algorithm, an NC can still be blocked if the target cell is not overloaded but one of its adjacent cells has reached its loading threshold. This is represented by the third term of (7). In the SD algorithm, if the NC is blocked at one cell which covers the SHZ, it still has a second chance in another cell that covers the SHZ. This is the reason why P_{bns} is used twice in (8).

4.5 Hot spot (non-homogeneous) traffic

In this section, we assume that the traffic load is non-homogeneous. Specifically, we assume that Cell 0 in Figure 4 is a hot spot. The traffic load for that cell is always higher than that of its adjacent cells. We apply a decomposition approach to model the system behavior under non-homogeneous traffic. The cluster of cells as shown in Figure 4 is decomposed into two SRNs, C_0 and C_1. SRN C_0 describes the behaviors of the MSs in Cell 0. SRN C_1 describes the behaviors of the MSs in cells 1 to 6. We also assume that the traffic loads in the boundary cells 7 to 18 are similar to those in cells 1 to 6.

4.5.1 Dropping probability

For non-homogeneous traffic, the dropping probabilities for the LA and the SD algorithms in C_0 and C_1 are denoted and calculated as following:

$$\begin{aligned}
P^{nh}_{dl0} &= \frac{(1-c_1\cdot P_4)\cdot \Lambda^3_e}{\Lambda_h} + \frac{P^f_q\cdot \lambda^d_h/6}{\lambda_t + \lambda^d_h}, \\
P^{nh}_{ds0} &= \frac{(1-P4)\cdot \Lambda^3_e}{\Lambda_h} + \frac{P^f_q\cdot \lambda^d_h/6}{\lambda_t + \lambda^d_h} + (1-c_0)\cdot P_{i1}, \\
P^{nh}_{dl1} &= \frac{(1-c_{nh}\cdot P_4)\cdot \Lambda^3_e}{\Lambda_h} + \frac{P^f_q\cdot \lambda^d_h/6}{\lambda_t + \lambda^d_h}, \\
P^{nh}_{ds1} &= \frac{(1-P4)\cdot \Lambda^3_e}{\Lambda_h} + \frac{P^f_q\cdot \lambda^d_h/6}{\lambda_t + \lambda^d_h} \\
&\quad +(1-c_1)\cdot (P_{i1}\cdot 5/6 + P_{i0}\cdot 1/6).
\end{aligned}$$

The probabilities that Cell 0 and the adjacent cells are not overloaded are respectively represented by c_0 and c_1. The parameters P_{i1} and P_{i0} are calculated in the same way as P_i in the previous section. Their numerical values are calculated from the SRN C_1 and SRN C_0, respectively. The weighted probability c_{nh} is calculated through $c_{nh} = c_0/6 + 5 \cdot c_1/6$.

4.5.2 Blocking probability

The blocking probabilities for the LA and the SD algorithms in C_0 and C_1 are respectively denoted and calculated as following:

$$
\begin{aligned}
P_{bl0}^{nh} &= (1-a) \cdot P_{bns} + a \cdot P_{bnc} + (1-c_1) \cdot \frac{\Lambda_n^c + \Lambda_n^s}{\lambda_n^c + \lambda_n^s}, \\
P_{bs0}^{nh} &= (1-a) \cdot P_{bns} \cdot P_{bns} + a \cdot P_{bnc}, \\
P_{bl1}^{nh} &= (1-a) \cdot P_{bns} + a \cdot P_{bnc} + (1-c_{nh}) \cdot \frac{\Lambda_n^c + \Lambda_n^s}{\lambda_n^c + \lambda_n^s}, \\
P_{bs1}^{nh} &= (1-a) \cdot P_{bns} \cdot P_{bns} + a \cdot P_{bnc}.
\end{aligned}
$$

4.5.3 Average dropping/blocking probability

To get an overall picture of the performance for a cluster of seven cells (cells 0 to 6 in Figure 4), an average dropping/blocking probability for the LA algorithm is calculated as following:

$$
P_{dl}^{nh} = (P_{dl0}^{nh} + 6 \cdot P_{dl1}^{nh})/7, \quad P_{bl}^{nh} = (P_{bl0}^{nh} + 6 \cdot P_{bl1}^{nh})/7.
$$

An average dropping/blocking probability for the SD algorithm can be obtained in a similar manner.

5 Numerical results and discussion

For the purpose of discussion, we make the following assumptions. The load threshold for a single cell is set to be 58. According to [26], $w_s = 1$ and $w_n = 0.47$. We set the queue length to be 2 and the number of soft guard channels (g) is set equal to 1. The ratio of the CZ area over the SHZ area is 1. The average inter-call time is assumed to be 2 minutes. The cell dwelling time is assumed to be exponentially distributed and its mean is set to be 4 minutes. The time out period is set to be 1 minute. Table 2 shows the state space for the SRN models presented in this paper.

Because it considers the traffic loads in both the target cell and the neighboring cells, the LA algorithm would block some new calls which would otherwise be

Models		No. of Tangible Markings	No. of Nonzero Transitions
Algorithm	Traffic		
LA	Homogeneous	1281	7240
	Non-homogeneous	3332	18960
SD	Homogeneous	1230	6944
	Non-homogeneous	3332	18960

Table 2
State space for the SRN models

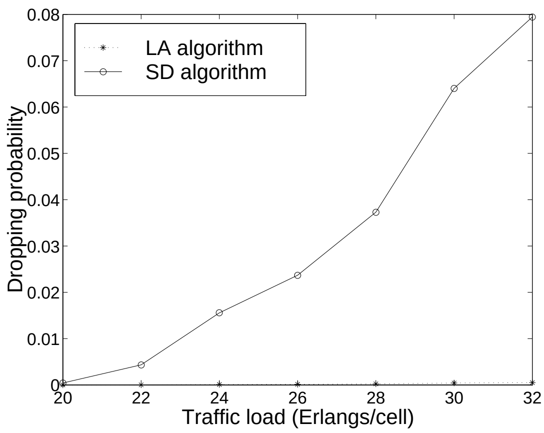

Fig. 8. Dropping probability under homogeneous traffic.

accepted in the SD algorithm. As a result, compared with the SD algorithm, the LA algorithm reduces the dropped calls at the expense of increasing the blocked calls. To evaluate the tradeoff between the dropped and blocked calls, we introduce two metrics, *increased dropping ratio* (r_d) and *increased blocking ratio* (r_b), which are defined as following:

$$r_d = \frac{P_d^{SD} - P_d^{LA}}{P_d^{LA}}, \quad r_b = \frac{P_b^{LA} - P_b^{SD}}{P_b^{SD}}$$

where P_d^{SD} and P_d^{LA} are dropping probabilities for the SD and the LA algorithms, respectively; P_b^{SD} and P_b^{LA} represent the blocking probability for the SD and the LA algorithms, respectively.

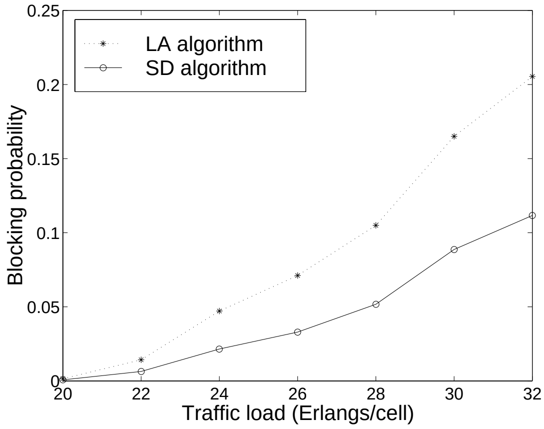

Fig. 9. Blocking probability under homogeneous traffic.

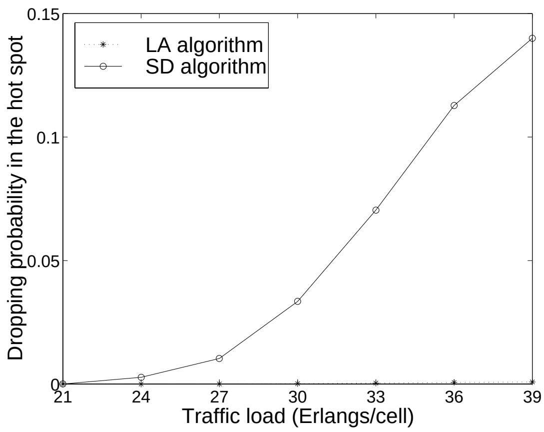

Fig. 10. Dropping probability in the hot spot.

5.1 Homogeneous scenario

In Figure 8, the dropping probabilities for different NC arrival loads under homogeneous traffic are shown. Compared with the SD algorithm, the LA algorithm can nearly eliminate the dropped calls. However, we do need to pay a price: an increased blocking probability for the LA algorithm as shown in Figure 9.

	Traffic Load in Erlangs						
Ratio	20	22	24	26	28	30	32
r_d $(\times 10^2)$	2.4	1.9	1.7	1.7	1.7	1.6	1.6
r_b	1.2	1.2	1.2	1.2	1.0	0.9	0.8

Table 3
Increased dropping/blocking ratios under homogeneous traffic

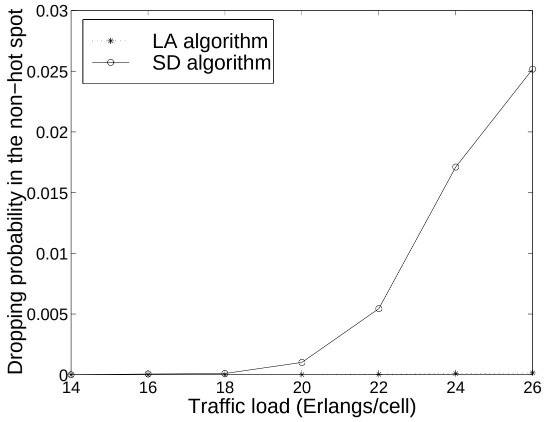

Fig. 11. Dropping probability in the non-hot spot.

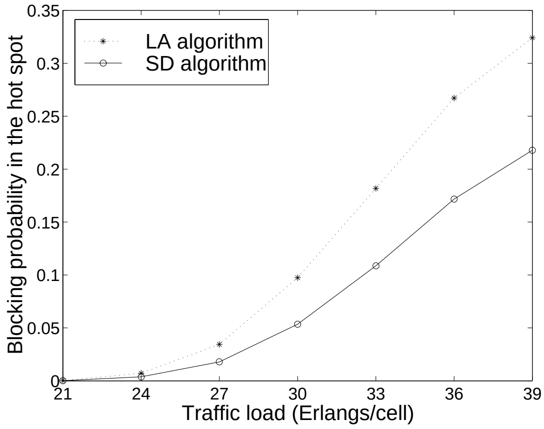

Fig. 12. Blocking probability in the hot spot.

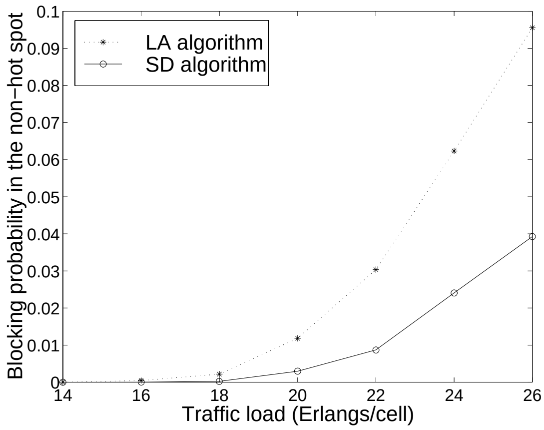

Fig. 13. Blocking probability in the non-hot spot.

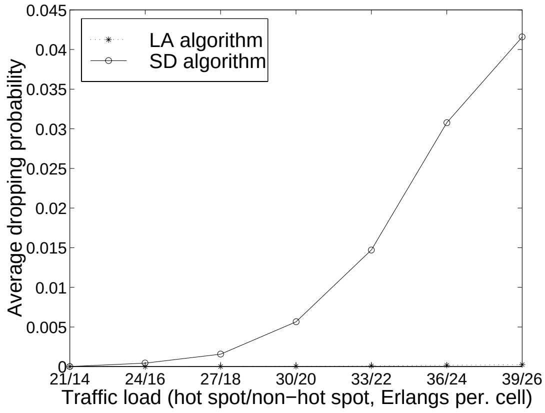

Fig. 14. Average dropping probability under non-homogeneous traffic.

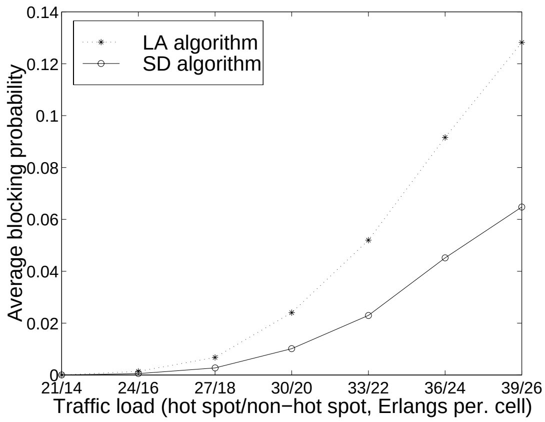

Fig. 15. Average blocking probability under non-homogeneous traffic.

	Traffic Load in Erlangs						
Ratio	21	24	27	30	33	36	39
r_d ($\times 10^2$)	3.4	4.1	2.4	2.1	1.8	1.7	1.6
r_b	1.0	0.9	0.9	0.8	0.7	0.6	0.5

Table 4
Increased dropping/blocking ratios for the hot-spot under non-homogeneous traffic

5.2 Hot spot (non-homogeneous) scenario

Next we assume that the traffic loads in Cell 0 (a hot spot) is 50% percent higher than that of its adjacent cells. Figures 10 and 11 show that the LA algorithm nearly eliminates the dropped calls in both the hot spot and its adjacent cells.

Figures 12 and 13 show that the LA algorithm increases the blocking prob-

Ratio	Traffic Load in Erlangs 14	16	18	20	22	24	26
r_d ($\times 10^2$)	8.3	15.3	1.9	1.1	1.8	1.9	1.6
r_b	82	17	8.5	3.0	2.5	1.6	1.4

Table 5
Increased dropping/blocking ratios for the non-hot spot under non-homogeneous traffic

ability vs. the SD algorithm under non-homogeneous traffic. For the average dropping/blocking probabilities, which are shown in Figures 14 and 15, we get the same conclusion as above.

Under homogeneous and non-homogeneous traffics, the increased dropping ratio (r_d) for the SD algorithm and the increased blocking ratio (r_b) for the LA algorithm are shown in Tables 3 to 5. From the data shown in Tables 3 and 4, we observe that the LA algorithm can decrease the dropping probability significantly while increasing the blocking probability at a relatively slow rate for homogeneous traffic and the hot-spot under non-homogeneous traffic. For the non-hot spots under non-homogeneous traffic (as shown in Table 5), when the traffic is low (below 16 Erlangs), the LA algorithm reduces the dropping probability and increases the blocking probability at nearly the same pace. For these traffic loads, since the traffic is light, the QoS can be guaranteed by using either the LA or the SD algorithm. When the traffic load is over 16 Erlangs, the LA algorithm can reduce the dropping probability significantly while increasing the blocking probability at a fairly small rate for the non-hot spots under non-homogeneous traffic.

6 Conclusion

In this paper, call admission control algorithms (the LA algorithms) with emphasis in reducing dropped calls are proposed. To give priority to soft handoff calls, we introduce the idea of 'soft guard channels'. SRN models are developed to analyze the performance of our algorithms.

The numerical results show that the LA algorithm can reduce the dropped calls to nearly zero. This is achieved by denying some of the new call arrivals. Consequently, the blocking probability in the LA algorithm is increased. Because of the capacity of the CDMA system is interference limited, when a new call is coming, we should base our admission decision on the loads of both the local and the adjacent cells. The conventional CAC methods only consider the local effect. Therefore their blocking probabilities are lower than the ones obtained from our algorithm. The drawback of the conventional methods is that they might cause dropped calls (or outage) in its adjacent cells when they have reached their saturate points.

Generally, there is a tradeoff between blocking probability and dropping probability. Our algorithm might be a good candidate for a system that requires a low dropping probability. To justify the gain we obtain through using the LA algorithm, two new metrics (increased dropping ratio and increased blocking ratio) are introduced. By using the LA algorithm, it is possible to reduce the dropped calls significantly by paying the price of increasing the blocking probability slowly.

References

[1] M. Ajmone-Marsan, D. Kartson, G. Conte, and S. Donatelli. *Modelling with Generalized Stochastic Petri Nets.* John Wiley & Sons, Inc., New York, NY, 1995.

[2] D. Aldous and L. Shepp. The least variable phase type distribution is Erlang. *Stochastic Models*, 3(3):467–473, 1984.

[3] G. Bolch, S. Greiner, H. de Meer, and K. S. Trivedi. *Queueing Networks and Markov Chains, Modeling and Performance Evaluation with Computer Science Application.* John Wiley & Sons, New York, NY, 1998.

[4] G. Ciardo, A. Blakemore, Jr. P. F. Chimento, J. K. Muppala, and K. S. Trivedi. Automated generation and analysis of Markov reward models using stochastic reward nets. In C. Meyer and R. Plemmons, editors, *Linear Algebra, Markov Chains and Queuing Models*, volume 48, pages 145–191. Springer-Verlag, 1993.

[5] G. Ciardo, J. K. Muppala, and K. S. Trivedi. SPNP Users Manual, Ver. 5.01. Technical report, Duke University, Durham, NC, 1998.

[6] G. Ciardo and K. S. Trivedi. A decomposition approach for stochastic reward net models. *Performance Evaluation*, 18(1):37–59, July 1993.

[7] Y. Fang, I. Chlamtac, and Y.-B. Lin. Call performance for a PCS network. *IEEE J. Select. Areas Commun.*, 15(8):1568–1581, Oct. 1997.

[8] G. Haring, R. Marie, R. Puigjaner and K. S. Trivedi. Loss formulae and their optimization for cellular networks. To appear in *IEEE Transactions on Vehicular Technology.*

[9] S. Hunter, T. Phillip, and K. S. Trivedi. Combined performance and availability analysis of a switched network application. In *Proceedings of the International Conference on Communications (ICC'97)*, Montréal, Québec, Canada, June 8-12 1997.

[10] O. C. Ibe, H. Choi, and K. S. Trivedi. Performance evaluation of client-server systems. *IEEE Transactions on Parallel and Distributed Systems*, 4(11):1217–1229, November 1993.

[11] Y. Ishikawa and N. Umeda. Capacity design and performance of call admission control in cellular CDMA systems. *IEEE J. Select. Areas Commun.*, 15(8):1627–1635, Oct. 1997.

[12] F. J. Jaimes-Romero, D. Muñoz-Rodríguez, C. Molina, and H. Tawfik. Modeling resource management in cellular systems using Petri nets. *IEEE Trans. Veh. Technol.*, 46(22):298–312, May 1997.

[13] C. Jedrzycki and V. C. M. Leung. Probability distribution of channel holding time in cellular telephony systems. In *Proceedings of IEEE 46th Vehicular Technology Conference (VTC'96)*, Atlanta, GA, USA, April & May 1996.

[14] J. C. Jr. Liberti and T. S. Rappaport. Analysis results for capacity improvements in CDMA. *IEEE Trans. Veh. Technol.*, 43(3):680–690, Aug. 1994.

[15] Z. Liu and M. E. Zarki. SIR-based call admission control for DS-CDMA cellular systems. *IEEE J. Select. Areas Commun.*, 12(4):638–644, May 1994.

[16] Y. Ma, J. J. Han, and K. S. Trivedi. A channel recovery method in TDMA wireless systems. In *Proceedings of the 50th IEEE International Vehicular Technology Conference (VTC'99–Fall)*, pages 1750–1754, Amsterdam, The Netherlands, Sep. 1999.

[17] Y. Ma, J. J. Han, and K. S. Trivedi. Channel allocation with recovery strategy in wireless networks. *European Transactions on Telecommunications (ETT)*, 11(4):395–406, July-August 2000.

[18] V. Mainkar and K. S. Trivedi. Sufficient conditions for existence of a fixed point stochastic reward net-based iterative models. *IEEE Trans. Software Engineering*, 22(9):640–653, Sep. 1996.

[19] C. Molina, N. Jain, and K. Basu. Performance model of cellular data on american systems. In *Proceedings of 1996 IEEE 46th Vehicular Technology Conference (VTC'96)*, Atlanta, GA, 28 April - 1 May, 1996.

[20] J. K. Muppala, A. S Sathaye, R. C. Howe, and K. S. Trivedi. Dependability Modeling of a Heterogeneous VAXcluster System Using Stochastic Reward Nets. In D. Averesky, editor, *Hardware and Software Fault Tolerance in Parallel Computing Systems*, pages 33–59. Ellis Horwood Ltd., 1992.

[21] G. Senarath and D. Everitt. Performance of handover priority and queuing systems under different handover request strategies for microcellular mobile communication systems. In *Proceedings of IEEE 45th Vehicular Technology Conference (VTC'95)*, pages 897–901, Stockholm, Sweden, June 1995.

[22] S.-L. Su, J.-Y. Chen, and J.-H. Huang. Performance analysis of soft handoff in CDMA cellular networks. *IEEE J. Select. Areas Commun.*, 14(9):1762–1769, Dec. 1996.

[23] N. D. Tripathi, J. H. Reed, and H. F. Vanlandingham. Handoff in cellular systems. *IEEE Personal Communications*, 5(6):26–37, Dec. 1998.

[24] K. S. Trivedi, Y. Ma, and J. J. Han. Performability analysis of fault tolerant RF link design in wireless communications networks. In *Proceedings of the 13th European Simulation Multiconference (ESM99)*, pages 33–40, Warsaw, Poland, June 1999.

[25] K. S. Trivedi, J. K. Muppala, S. P. Woolet, and B. R. Haverkort. Composite performance and dependability analysis. *Performance Evaluation*, 14(3-4):197–215, Feb. 1992.

[26] A. J. Viterbi. *CDMA: Principles of Spread Spectrum Communication*. Addison-Wesley, Reading, MA USA, 1995.

[27] C.-Y. Wang, D. Logothetis, and K. S. Trivedi. Transient behavior of atm networks under overloads. In *Proceedings of the IEEE INFOCOM 96*, pages 978–985, San Francisco, CA, USA, March 1996.

[28] C.-Y. Wang and K. S. Trivedi. Integration of specification for modeling and specification for system design. In M. Ajmone Marsan, editor, *Lecture Notes in Computer Science*, volume 691, pages 473–492. Springer-Verlag, Heidelberg, 1993.