Call Admission Control for Reducing Dropped Calls in Code Division Multiple Access (CDMA) Cellular Systems

Yue Ma  
Global Software Division  
Motorola Inc.  
1303 E. Algonquin Rd.,  
IL01/Anx 2  
Schaumburg, IL 60196-1065  
Yue.Ma@motorola.com

James J. Han  
Global Software Division  
Motorola Inc.  
1303 E. Algonquin Rd.,  
IL01/Anx 2  
Schaumburg, IL 60196-1065  
James.Han@email.mot.com

Kishor S. Trivedi  
Department of Electrical and Computer Engineering  
1112 Hudson Hall  
Duke University, Box 90291  
Durham, NC 27708-0291  
KST@EE.DUKE.EDU

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.

Keywords— call admission control, CDMA, interference, soft guard channel, Markov reward models, stochastic reward net (SRN).

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 [23]. 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 [9], [12], [18] 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 SD 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 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 LA 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 inter-
fERENCE limited. There are no physical resources which can be
used as guard channels. Instead, a certain amount (either inte-
gral 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 II, some of
the related features of a CDMA cellular system are described.
In Section III, we present our CAC algorithms, stochastic reward
nets (SRNs) are developed in Section IV. A brief introduction of SRN is first given at the begin-
ing of that section. In Section V, the performance of the LA
and the SD algorithms are compared. Finally, conclusions are
made in Section VI.

II. SYSTEM DESCRIPTION

A. 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 fre-
quency. 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 conven-
tional cellular network, the MS breaks the communication link
with the old base station (BS) before establishing a new com-
munication link with the new BS. This kind of handoff process
(break before make) is generally experienced with a brief in-
terception of communication and is commonly referred as hard
handoff.

In an IS-95 [19] 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 commu-
nication 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 ser-
vice reliability is substantially improved through soft handoff.

B. CDMA Cellular System Model

Assume that a target cell in a wireless network can be parted-
tioned 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 neigh-
borhood zone (NZ). Fig. 1 shows a concentric geometry used
for the purpose of illustrating the relationship among the three
zones. In [11], similar concentric geometry approaches are used
for analyzing the performance of cellular networks.

![Fig. 1. A concentric geometry for a target cell and its neighbor-
hood area.]

When an MS is in soft handoff phase, we assume that at most
two base stations are in diversity reception. New call inter-
arrival times are assumed to be independent and exponentially
distributed [10], with rate $\lambda_n$. 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 $\lambda_n^C$ or $\lambda_n^S$ is respectively the new call arrival rate in CZ
and SHZ and $a = \frac{\text{area of CZ}}{\text{area of target cell}}$. Suppose
that the mean target cell dwelling time is $\mu_d$, 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], [8], the call holding
times are assumed to be exponentially distributed.

III. 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 neigh-
borhood. 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.
Fig. 2. CAC algorithm for the new call arrivals in the target cell.

A. CAC Algorithm for New Calls

Fig. 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:

$$ELT = k_c + w_x \cdot k_x + w_n \cdot k_n$$

where $k_c$ ($k_x$) 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; $w_x$ and $w_n$ are the weights. 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 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 [18], 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.

B. CAC Algorithm for Soft Handoff Calls

The CAC algorithm for soft handoff calls is shown through Fig. 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.

C. 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 [20], [22] 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 [17] 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 Fig. 2 is denoted as $TN$ and the threshold in Fig. 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.

IV. SRN MODELS FOR THE CAC ALGORITHMS

A. Introduction to SRN

Stochastic reward net (SRN) [4] is an extension of Petri net (PN) [16], 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 ($ERG$) 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 $ERG$ can be reduced to a homogeneous continuous time Markov chain (CTMC) [1].

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], [21], steady state and/or transient analysis of the MRM produces the required measures of the original SRN. In this paper, we use the tool SPNP [3], [6] to specify and solve the SRN models.

B. Model Description

In this section, we construct SRN models in accordance with the CAC algorithms described in Section III. Because of the interaction among the cells, an SRN model for even a cluster of cells (Fig. 4) would instigate a huge state space for the underlying Markov reward process. Depending on the nature 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 [15]; hierarchical approach [13] and decomposition approach [5]. In this paper, our SRN models are based on a decomposition approach. Fig. 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 Fig. 5.

In Fig. 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 $t^n_1$ and $t^n_2$, respectively. Whether the target load of the target cell is under the threshold or not is determined by the enabling functions for transitions $t^n_1$ and $t^n_2$. If the adjacent cells are not over-
loaded, 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 \( t_5^d \) 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 \( t_2^d \) and \( t_2^s \). With regard to the other cell which covers the SHZ, transition \( t_2^d \) also represents a handoff arrival for that cell; while transition \( t_2^s \) 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 \( t_5^d \).
- The user moves to an 'outside' adjacent cell, with rate \( \lambda_A^d \).
- The call is normally terminated, with rate \( \lambda_t \). In Fig. 5, these last two transitions are combined into one transition \( t_2^d \), with \( \lambda_2^d = \lambda_A^d + \lambda_t \).

For a handoff arrival, represented by transition \( t_4 \), the call can be accepted immediately through transition \( t_4^d \), 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 Fig. 5 describes a queue with time out. The time out event is modeled by a three stage Erlang distribution [2], [24]. Transition \( t_4^e \), 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 \( t_4 \) to \( t_4^d \)) 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 \( t_4^d \) through \( t_4^s \). 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 \( t_4^d \) through \( t_4^s \). 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 \( t_5^d, t_5^e, t_6^d, t_6^e, t_7^d, t_7^e, t_8^d, t_8^e \), and \( t_9^d \) 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 Fig. 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 Fig. 5, immediate transitions \( t_4 \) through \( t_9 \) and the places \( T_1 \) and \( T_2 \) are removed from Fig. 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 \( t_4 \).

C. 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.
- \( \lambda_h \): handoff arrival rate.

To capture this dynamic behavior, a fixed-point iteration scheme [14] is applied to determine the above parameters. The values of these parameters are calculated as following:

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

\[
c = \sum_{j \in \Omega} r_{2j} \cdot \eta_j(k_n, c, \lambda_h), \tag{2}\n\]

\[
\lambda_h = (\Lambda_2^d + \Lambda_2^e + \Lambda_2^2 + \Lambda_2^3) \cdot \Lambda_d^d + \Lambda_d^e, \tag{3}\n\]

where \( \Omega \) is the set of tangible markings in the SRN models and \( \pi_j \) is the steady state probability of marking \( j \). We use \( \Lambda \) to denote the throughput of a transition. For example, \( \Lambda_2^d \) is the throughput of transition \( t_2^d \). In (2), the reward rate assignment \( r_{2j} \) is of the following form.
\[ r^4_t = \begin{cases} 
1 & \text{if } (\#[CZ] + w_z \cdot \#[SHZ] + w_n \cdot k_n) \leq T - g, \\
0 & \text{otherwise.}
\end{cases} \]

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. Fig. 7 shows the fixed-point iteration algorithm we use for solving the SRN in Fig. 5 under homogeneous traffic. According to Theorem 2 in [14], 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 are under investigation.

```plaintext
/* initialization, \( k_h^{(0)}, c_h^{(0)}, \lambda_h^{(0)} \) are initial values. */
1. \( k_h^{(1)} = k_h^{(0)}; \)
2. \( c_h^{(1)} = c_h^{(0)}; \)
3. \( \lambda_h^{(1)} = \lambda_h^{(0)}; \)
4. repeat
5. solve the SRN in Fig. 5;
6. compute \( k_h^{(1+1)}, c_h^{(1+1)} \) and \( \lambda_h^{(1+1)} \)
   according to (1), (2) and (3), respectively;
7. \( \delta_1 \leftarrow k_h^{(1+1)} - k_h^{(1)} / k_h^{(0)}; \)
8. \( \delta_2 \leftarrow c_h^{(1+1)} - c_h^{(1)} / c_h^{(0)}; \)
9. \( \delta_3 \leftarrow \lambda_h^{(1+1)} - \lambda_h^{(1)} / \lambda_h^{(0)}; \)
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 Fig. 5 under homogeneous traffic.
```

D. 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.

D.1 Enabling function

The NC and SHC arrivals are accepted only if the CAC conditions are satisfied. In an SRN, this is achieved through enabling functions and the probabilities associated with the transitions.

At transition \( t^4_n \), a new call is accepted under the condition:

- \( (\#[CZ] + w_z \cdot \#[SHZ] + w_n \cdot k_n) \leq T - w_x \cdot g - 1. \)

This condition is associated with the enabling function of \( t^4_n \) 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 \( t^6_n, t^8_n, \) and \( t^1_t \) through \( t^2_t \).

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

<table>
<thead>
<tr>
<th>Transition</th>
<th>Enabling Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>( t^6_n )</td>
<td>( #[CZ] + w_z \cdot #[SHZ] + w_n \cdot k_n \leq T - w_x \cdot g - 1 )</td>
</tr>
<tr>
<td>( t^8_n )</td>
<td>( #[CZ] + w_z \cdot #[SHZ] + w_n \cdot k_n \leq T - w_x \cdot (g + 1) )</td>
</tr>
<tr>
<td>( t^1_t )</td>
<td>( #(Q) \leq 0 )</td>
</tr>
<tr>
<td>( t^2_t )</td>
<td>( #(Q) = 0 )</td>
</tr>
<tr>
<td>( t^3_t )</td>
<td>( #(Q) = 0 )</td>
</tr>
<tr>
<td>( t^4_t )</td>
<td>( #(Q) = 0 )</td>
</tr>
</tbody>
</table>

Table I

**Enabling Functions for the LA Algorithm**

D.2 Dropping probability

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

\[
P_{dl}^h = \frac{(1 - c \cdot P_2) \cdot \Lambda_2^h + P_1 \cdot \Lambda^2_2 / 6}{\Lambda_2 + \Lambda^2_2 / 6},
\]

\[
P_{dl}^h = \frac{(1 - P_1) \cdot \Lambda_2 + P_1 \cdot \Lambda^2_2 / 6 + (1 - c) \cdot P_4}{\Lambda_2 + \Lambda^2_2 / 6},
\]

where

\[
P_1 = \frac{\Lambda_2 + \Lambda^2_2}{\Lambda_2 + \Lambda^2_2 / 6}.
\]

In the above equations, \( P_2 \) is the probability that the enabling function for transition \( t^4_n \) is enabled, \( P_4 \) 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 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).
D.3 Blocking probability

Under homogeneous traffic, the blocking probability for the LA and the SD algorithms is denoted as \( P^h_{b0} \) and \( P^h_{b0} \) respectively. They are calculated according to the following formulas:

\[
P^h_{b0} = (1 - a) \cdot P_{bne} + a \cdot P_{bnc} + (1 - c) \cdot \frac{\lambda^*_n}{\lambda^* + \lambda_n},
\]

\[
P^h_{b0} = (1 - a) \cdot P_{bne} + a \cdot P_{bnc},
\]

where \( P_{bnc} \) and \( P_{bne} \) 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 \( P_{bne} \) is used twice in (8).

E. Hot spot (non-homogeneous) traffic

In this section, we assume that the traffic load is non-homogeneous. Specifically, we assume that Cell 0 in Fig. 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 Fig. 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.

E.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:

\[
P^h_{d0} = (1 - c_0 \cdot P_{t0} - \Lambda^2_0 / \Lambda_0 + \Lambda_A / \Lambda_0 + \Lambda_0 / \Lambda_A) \cdot P_{b0} + (1 - c_0) \cdot P_{t0},
\]

\[
P^h_{di} = (1 - c_n \cdot P_{ti} - \Lambda^2_n / \Lambda_n + \Lambda_A / \Lambda_n + \Lambda_n / \Lambda_A) \cdot P_{b0} + (1 - c_0) \cdot P_{t0},
\]

\[
P^h_{d1} = (1 - c_n \cdot P_{ti} - \Lambda^2_n / \Lambda_n + \Lambda_A / \Lambda_n + \Lambda_n / \Lambda_A) \cdot P_{b0} + (1 - c_0) \cdot P_{t0}.
\]

The probabilities that Cell 0 and the adjacent cells are not overloaded are respectively represented by \( c_0 \) and \( c_1 \). The parameters \( P_{t0} \) and \( P_{t1} \) are calculated in the same way as \( P_t \) 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 \( c_{nh} = c_0 / 6 \cdot 5 + c_1 / 6 \).

E.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 follow:

\[
P^{h/A}_{b0} = (1 - a) \cdot P_{bne} + a \cdot P_{bnc} + (1 - c_0) \cdot \frac{\Lambda^*_n + \Lambda^*_n}{\Lambda^*_n + \Lambda_n},
\]

\[
P^{h/A}_{b0} = (1 - a) \cdot P_{bne} \cdot P_{bnc} + a \cdot P_{bnc},
\]

\[
P^{h/A}_{b0} = (1 - a) \cdot P_{bne} + a \cdot P_{bnc} + (1 - c_{nh}) \cdot \frac{\Lambda^*_n + \Lambda^*_n}{\Lambda^*_n + \Lambda_n},
\]

\[
P^{h/A}_{b0} = (1 - a) \cdot P_{bne} \cdot P_{bnc} + a \cdot P_{bnc}.
\]

E.3 Average dropping/blocking probability

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

\[
P^{h/A}_{d0} = (P^{h/A}_{d0} + 6 \cdot P^{h/A}_{d1}) / 7, \quad P^{h/A}_{b0} = (P^{h/A}_{b0} + 6 \cdot P^{h/A}_{b1}) / 7.
\]

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

V. 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 [23], \( w_s = 1 \) 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 II shows the state space for the SRN models presented in this paper.

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Traffic</th>
<th>No. of Tangible Markings</th>
<th>No. of Nonzero Transitions</th>
</tr>
</thead>
<tbody>
<tr>
<td>LA</td>
<td>Homogeneous</td>
<td>1281</td>
<td>7240</td>
</tr>
<tr>
<td>Non-homogeneous</td>
<td>3332</td>
<td>18960</td>
<td></td>
</tr>
<tr>
<td>SD</td>
<td>Homogeneous</td>
<td>1230</td>
<td>6944</td>
</tr>
<tr>
<td>Non-homogeneous</td>
<td>3332</td>
<td>18960</td>
<td></td>
</tr>
</tbody>
</table>

| TABLE II | STATE SPACE FOR THE SRN MODELS |

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 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, \textit{increased dropping ratio} (\( \tau_d \)) and \textit{increased blocking ratio} (\( \tau_b \)), which are defined as following:

\[
\tau_d = \frac{P^s_{d} - P^l_{d}}{P^l_{d}}, \quad \tau_b = \frac{P^s_{b} - P^l_{b}}{P^l_{b}}
\]

where \( P^s_{d} \) and \( P^l_{d} \) are dropping probabilities for the SD and the LA algorithms, respectively; \( P^s_{b} \) and \( P^l_{b} \) represent the blocking probability for the SD and the LA algorithms, respectively.
A. Homogeneous scenario

In Fig. 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 Fig. 9.

<table>
<thead>
<tr>
<th>Ratio</th>
<th>20</th>
<th>22</th>
<th>24</th>
<th>26</th>
<th>28</th>
<th>30</th>
<th>32</th>
</tr>
</thead>
<tbody>
<tr>
<td>$r_d$ ($\times 10^3$)</td>
<td>2.4</td>
<td>1.9</td>
<td>1.7</td>
<td>1.7</td>
<td>1.6</td>
<td>1.6</td>
<td>1.6</td>
</tr>
<tr>
<td>$r_b$</td>
<td>1.2</td>
<td>1.2</td>
<td>1.2</td>
<td>1.2</td>
<td>1.0</td>
<td>0.9</td>
<td>0.8</td>
</tr>
</tbody>
</table>

TABLE III
INCREASED DROPPING/BLOCKING RATIOS UNDER HOMOGENEOUS TRAFFIC

B. 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. Figs. 10 and 11 show that the LA algorithm nearly eliminates the dropped calls in both the hot spot and its adjacent cells.

Figs. 12 and 13 show that the LA algorithm increases
the blocking probability vs. the SD algorithm under non-homogeneous traffic. For the average dropping/blocking probabilities, which are shown in Figs. 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 III to V. From the data shown in Tables III and IV, 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 V), 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.

VI. 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


