An Efficient Distributed Channel Allocation Strategy Based on A Threshold Scheme for Mobile Cellular Networks

Yongbing Zhang[†], Sajal K. Das[‡] and Xiaohua Jia[*]

[†]Institute of Policy and Planning Sciences, University of Tsukuba, Ibaraki 305-8573, Japan
[‡]Department of Computer Science and Engineering, University of Texas at Arlington, Texas 76019-0015, USA
[*]Department of Computer Science, City University of Hong Kong, Kowloon, Hong Kong

Abstract—

We propose a distributed channel allocation algorithm based on a threshold scheme, called D-CAT, for cellular mobile networks. The algorithm employs two thresholds to determine whether a cell is heavy, i.e., overloaded, and the optimal number of free channels as well as the cell(s) from where it needs to import in order to satisfy its channel demand. Simulation experiments and analyses show that the D-CAT algorithm incurs lower overhead for channel allocation and is more efficient in terms of channel utilization than other distributed channel allocation algorithms. It also outperforms other centralized and distributed algorithms in terms of call blocking probability.

I. INTRODUCTION

Various technologies such as Frequency Division (FD), Time Division (TD), and Code Division (CD) [1] are used in mobile cellular networks to utilize the radio spectrum. The radio spectrum is further divided into channels to serve different calls and many schemes have been proposed to allocate channels to the cells such that the available channels are efficiently used and thus the channel reuse is maximized [2], [3], [4], [5], [6]. The performance metric used for measuring the efficiency of a channel allocation scheme is the call blocking probability, i.e., the sum of the probabilities of new call blocking as well as forced termination.

Channel allocation strategies can be broadly classified into two categories: fixed [7] and dynamic [8]. A fixed allocation (FA) strategy is to allocate a fixed set of channels to each cell permanently. It is simple to implement, but cannot reallocate channels at run-time. In contrast, a dynamic allocation (DA) strategy is to allocate the channels in the system dynamically. It tends to be more efficient than an FA strategy in conditions of light, non-homogeneous, and time-varying traffic but at the cost of high implementation overhead.

The issue of who plays a key role in a channel allocation decision is very important. In a centralized channel allocation algorithm [9], [3], [6], a mobile switching center (MSC) plays the key role and accomplishes the channel allocation. The disadvantage of centralized algorithms is that the MSC may be overloaded and the failure of the MSC makes the whole system down. In a distributed channel allocation algorithm [2], [10], [11], [9], [4], on the other hand, each base station (BS) at a cell plays the key role in a channel allocation decision and is capable of running the channel allocation algorithm. The main advantage of a distributed algorithm is its high reliability and scalability but its main disadvantage is its high implementation cost, i.e., high overhead cost for message exchanges among the cells, distributed time clock and resource management, etc.

In this paper, we propose a distributed dynamic channel allocation algorithm, called D-CAT, based on a threshold scheme. The D-CAT algorithm employs two thresholds: (i) a heavy threshold used for determining whether a cell is heavy, i.e., overloaded, and for triggering the channel allocation algorithm; and (ii) a target threshold used for indicating the target number of free channels that a heavy cell intends to acquire. Using the proposed threshold scheme, a heavy cell can determine the optimal number of free channels it needs to acquire and from where it should import the channels. As a result, the overhead cost for channel allocation is minimized and the available channels among the cells are balanced.

II. SYSTEM MODEL

As shown in Figure 1, the geographical area is typically divided into hexagonal cells in a mobile cellular network. Each cell is served by a base station (BS) and the mobile users communicate through wireless links using radio channels. A number of cells (or BSs) are linked to a mobile switching center (MSC) through dedicated wire-line links. Each MSC is linked to the fixed telephone network (e.g., PSTN and ISDN) again through a wire-line link and acts as a gateway of the cellular network to the fixed backbone network [12].

In our model, the system has a total of S distinct channels which are initially assigned to the cells in the same way as in the fixed channel assignment scheme. However, no channel belongs to a specific cell permanently. Channels initially assigned to a cell are called the origin channels to the cell. Figure 1 shows an example of the initial state of the system where the alphabets a, b, c, ... on the cells denote different sets of channels and the set of cells using distinct sets of channels is 7. The cells with the same alphabet are assigned the same set of channels. Any channel can be reassigned to any other cell if necessary provided that the same channel is used farther enough than the reuse distance, the minimum distance at which the same channel can be reused without interference. For example, all the cells with the same alphabets in Figure 1 are assigned the same set of channels with the minimum reuse distance. A cell holds the channels assigned to it as its own property and has the right to control them freely. A set of cells in the system forms a group, N_i, so that each cell in N_i is located within the minimum reuse distance related to the center of the group, say cell i, as shown in Figure 2.

A newly incoming call or a handoff call from an adjacent cell will be assigned a free channel immediately if there are any available. When an active call traverses the boundary of two cells an inter-cell handoff occurs and the call releases the serving channel in the original cell and is reassigned a new channel at the adjacent destination cell. A cell may enforce an active call to release its serving channel and reassign it with a new one in the same cell. This process is called an intra-cell handoff.

Two thresholds in D-CAT, a heavy and a target thresholds,

0-7803-7208-5/01/$17.00 (C) 2001 IEEE

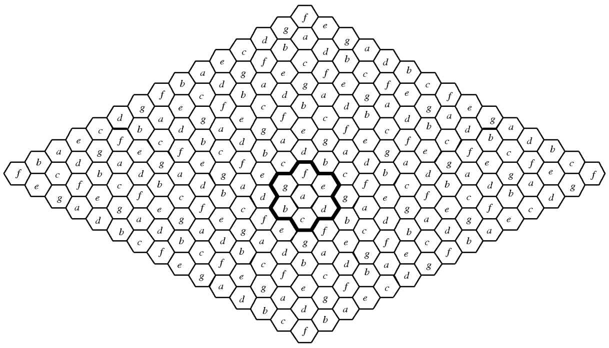


Fig. 1. Cellular System

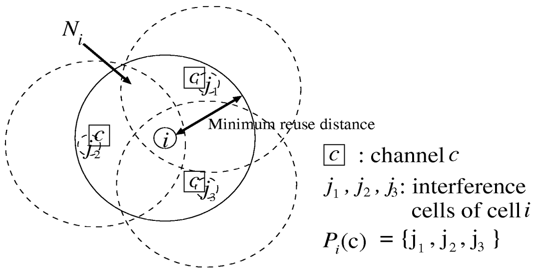


Fig. 2. Interference cells

are denoted by T^h and T_i^t, respectively. Cell i is called overloaded or heavy, if the number of channels, v_i, available at cell i is less than or equal to T^h. A cell intends to import free channels if it becomes heavy but will do its best to respond to channel requests from other cells if it has plenty of free channels.

The D-CAT algorithm is triggered by cell i in the event of a new call arrival when $v_i \cdot\ T^h$. Typical values of T^h are 0 and 1, and determined according to the necessity of the channel allocation policy. If $T^h = 1$, a cell can still assign a channel to a newly arriving call while attempting to import free channels. On the other hand, when $T^h = 0$, a cell attempts to import free channels only if it has no free channels anymore. The target threshold, T_i^t, indicates the target number of free channels that cell i intends to acquire. The value of T_i^t is determined by the average number of channels available at a cell in the group N_i as follows.

$$T_i^t = \left\lfloor \frac{\sum_{j \in N_i, j \neq i} v_j}{|N_i|} + 0.5 \right\rfloor, \tag{1}$$

where v_j is the number of channels available at cell j. If $T_i^t = 0$ then let $T_i^t = 1$. Note that each cell in the system has the same T^h value but a distinct T_i^t value.

A heavy cell attempting to obtain free channels is called an importer. On the other hand, a cell that can provide free channels is called an exporter. A cell can reserve few channels (typically 1 channel) for the next newly arriving calls and assign them to the new arrival calls while processing requests from importers. When a heavy cell (importer) attempts to import any reserved channel, it must confirm its availability. Before approving a request for a reserved channel from an importer, an exporter can assign the reserved channel to a newly arriving call at the exporter.

The following notations are used in this paper.

S: set of channels used in the system
C_i: set of cells as candidates for channel importing at cell i
V_i: set of channels available at cell i
v_i: number of channels available at cell i, $v_i = |V_i|$
U_i: set of channels used in cell i
u_i: number of channels used in cell i, $u_i = |U_i|$
N_i: set of the neighboring cells within the interference distance of cell i
R_{ij}: set of channels imported by cell i from cell j
R'_{ij}: set of confirmed channels exported from cell j to cell i
T^h: heavy threshold
T_i^t: target threshold at cell i
r_i: number of channels needing to import for cell i, $r_i = T_i^t - v_i$
$P_i(c)$: set of cells that own channel c in N_i

III. THE D-CAT ALGORITHM

Each cell in D-CAT maintains a group of channels and treats the channels it holds as its own property. Furthermore, if a cell obtains any free channels from its interfering neighbors, it also keeps them as its property. It is therefore not a borrower-lender relation between a channel requester and supplier but instead an importer-exporter relation.

When a new call arrives at a heavy cell, the D-CAT algorithm is activated requesting its neighboring interference cells for help, and attempts to import sufficient free channels to satisfy its demand. The messages transmitted between cell i (channel importer) and cell j (possible channel exporter) are classified into four categories as follows.

- request message, $request(i)$: Message sent by importer i to all the neighboring cells in N_i to request free channels.
- reply message, $reply(j, V_j, U_j)$: Message from cell $j \in N_i$ responding to importer i. The message also includes the information on the reserved channels in cell j.
- inform message, $inform(i, R_{ij})$: Message sent by importer i to the exporters and the other cells in N_i to inform them about its channel acquisition decision. The message also includes the requests of the reserved channels if any.
- confirm message, $confirm(j, R'_{ij})$: Message sent by exporter j to importer i to inform it the availability of the requested channels that have been reserved at exporter j. Exporter j can still assign the reserved channels to new arrival calls before sending the confirm message back to importer i.

Each message contains a timestamp which equals the time at which the message was sent. Using the timestamps of the messages, the time of messages transmitted among the cells can be synchronized.

The D-CAT algorithm consists of three components – channel import component, channel export component, and channel selection component. The first one is activated by a heavy cell needing free channels and works as the client in the channel acquisition process. The second one, on the other hand, is always active at each cell and ready to perform as the server to receive channel requests from clients. The last one is used for selecting appropriate channels to import and for assigning and reassigning channels in a cell. In the following sub-sections, the three components of D-CAT algorithm and its deadlock freedom are described.

A. Channel Import Component

When a cell becomes heavy, each new arriving call at the cell makes channel requests to its interference neighbors for free channels. The channel acquisition algorithm can be described as the following seven steps.

1. When cell i becomes heavy, i.e., $v_i \cdot \; T^h$, it sends a request message, $request(i)$, to all of its interference neighbors in N_i. When cell i receives a request from another cell with a larger timestamp, it postpones the response. Otherwise, it replies the request immediately.
2. After cell i has received all the reply messages from its neighbors, it calculates T_i^t and r_i.
3. Seek for the unused channels in N_i, i.e., $S - \cup_{j \in N_i} \big(V_j \cup U_j\big) - \big(V_i \cup U_i\big)$. Obtain the set of free channels, R_0, so that $|R_0| = \min\big\{r_i, |S - \cup_{j \in N_i} \big(V_j \cup U_j\big) - \big(V_i \cup U_i\big)|\big\}$. Stop the algorithm if the request is satisfied. Otherwise, go to the next step.
4. Treat the reserved channels in the neighboring cells as the used ones and search for free channels according to the following sub-steps.
   (a) Search for cells $j \in N_i$ such that $v_j > T_i^t$, and denote the set of these cells by C_i.
   (b) Select a channel c that belongs to cell $j \in C_i$ and $|P_i(c)| = 1$. Add channel c to the import channel set, R_{ij}. Delete cell j from C_i if the condition of $v_j > T_i^t$ is violated. Repeat this process until the request is satisfied, there are no more appropriate channels, or C_i becomes empty.
   (c) Select a channel c that belongs to cell $j \in P_i(c)$ and $P_i(c) \subset C_i$. That is, find a channel c belonging to cells j that satisfies the condition of $v_j > T_i^t$. Add channel c to R_{ij}. Delete cell j from C_i if the condition of $v_j > T_i^t$ is violated. Repeat this process until the request is satisfied, there are no more appropriate channels, or C_i becomes empty.
   (d) Search for a channel c such that c belongs to cell $j \in P_i(c)$ and the following inequality is satisfied.

   $$\max\big\{\min_j(v_j, j \in P_i(c))\big\} - 1 \geq v_i + \cup_{j \in N_i} |R_{ij}| + |R_0| + 1.$$

   Add channel c to R_{ij}. Repeat this process until the request is satisfied, or there are no more appropriate channels.
5. If cell i still needs more free channels, treat the reserved channels in the neighboring cells as free channels this time and repeat steps 4(a)–4(d).
6. Mark the channels in R_{ij} that are reserved at cell j and send an inform message, $inform(i, R_{ij})$, to each channel exporter j to inform which free channels are imported and which reserved channels are requested. Wait for the confirmation of the availability for the reserved channels in R_{ij} but the other free channels are ready for immediate use.
7. After receiving the confirm message, $confirm(j, R'_{ij})$, from exporter j, make the confirmed channels available and discard the other ones.

B. Channel Export Component

Channel requests arrived at cell j are queued in a request queue based on the timestamps of the requests and then processed sequentially. When cell j receives a request from cell i, it processes the request according to the following four steps.

1. If there are no requests under processing, go to step 2. Otherwise, compare the timestamps of the requests. If the newly incoming request from cell i has a smaller timestamp than the request from cell k under processing, then the request from cell k will be aborted and put into the request queue and then go to step 2.
2. Reply cell i with a reply message, $reply(j, V_j, U_j)$, and wait until the inform message arrives if cell j is not heavy. If cell j is heavy it has no need to wait but can continue its own processing.
3. When cell j receives an inform message, $inform(i, R_{ij})$, it locks the requested channels. If any channels in R_{ij} are the reserved ones but still available, cell j will lock these channels and add them to R'_{ij}. It then selects new reserved channels if cell j still has free channels. Cell j can however assign a reserved channel to a new call arrived locally before sending the confirm message to cell i.
4. Send cell i a confirm message, $confirm(j, R'_{ij})$, to inform the availability of the reserved channels.

C. Channel Selection Component

The channel reuse pattern with an irregular channel assignment scheme may not be optimized, since the irregular channel assignment to a cell may make channels be reused with a distance longer than the minimum reuse distance in order to avoid interference. This degrades the channel utilization of the system.

The channel selection component consists of two subcomponents: one used for importing free channels from exporters and the other used for assigning and reassigning channels in a cell. When a heavy cell finds multiple candidates in any stage of the import process, it prioritizes the candidates depending on their identifiers. A channel with an identifier which is the same as or nearer to any identifiers of the channels initially assigned to the importer has a higher priority. The best candidate will be chosen from these candidates. For example, let cell i be initially assigned with three channels of 5, 6, and 7, and now have only one channel 5. If cell i needs to import free channels and has found four candidates, 1, 4, 6, and 9, then it attempts to import these channels with a priority of 6, 4, 9, and 1.

Channel assignment and reassignment in a cell are performed according to the channel origins. Channels except the original channels at a cell are those imported from its interfering neighbors. When a new call arrives at a cell, an original channel is assigned to the call with the highest priority. For two original channels, the one with the smaller identifier has a higher priority. The imported channels are prioritized according to how far their identifiers from those of the original channels. The nearer the identifier of an imported channel to that of any original channel the higher the priority of the imported channel. For two channels with the same priority, the one with the smaller identifier has a higher priority. Furthermore, if an original channel is released while an imported channel is serving an active call, an intra-cell handoff is performed so that the original channel is reassigned to the active call. A free imported channel with a higher priority will also cause an intra-cell handoff if another imported channel with a lower priority is serving an active call.

D. Deadlock Freedom of D-CAT algorithm

Since the cells can send messages autonomously and concurrently, the synchronization problem of time for requests is the same as that in a distributed system. Assuming that the communication link is reliable and each message contains a timestamp showing when it was sent, and that the messages sent by a cell arrive at a cell in the order in which they are sent, the request messages originating from different cells can be totally ordered by their timestamps [13]. Since the time ordering of requests are known by all the cells, there is no loop for delaying reply messages among the cells, and the cell whose request has the lowest timestamp can always receive all of reply messages from its neighbors. After determining the imported channels, it sends an inform message to all of its neighbors. An exporter receiving an inform message can decide immediately whether it approves the requested channels or rejects some of them, and send a confirm message back to the importer. The importer determines how many channels it can import successfully and then processes the postponed replies. Since the operation at each step in the process of acquiring free channels is time-limited, no request will wait forever.

IV. PERFORMANCE EVALUATION

The performance of D-CAT is evaluated with respect to two metrics: the implementation cost and the call blocking probability. The number of messages transmitted between the BSs and the delay of message transmission are taken into account in the implementation cost. Two distributed algorithms, D-LBSB [9] and an algorithm [2] referred to as D-ES in this paper, are chosen to compare with D-CAT. An efficient centralized algorithm [6], referred to as CAT in this paper, is also compared with D-CAT.

A. Implementation Cost Comparison

Let the total number of cells in the system is denoted by N. The message delay between the BSs and between a BS and the MSC is fixed to be δ. The postponed response delay experienced at a channel exporter in D-ES and D-CAT is denoted by δ_d. In D-ES, n_p denotes the number of interference primary neighbors of a cell, m denotes the conflict rate, and n_u denotes the update message. Since the execution time of a channel allocation algorithm is much smaller than the message delay, it is not taken into account in the comparison. In order not to bring any bias to the comparison results for the D-LBSB and CAT algorithms, the case of locking only three co-channels for each lender cell is taken into account in these two algorithms. It is also assumed that a heavy cell needs X channels and each channel exporter can offer only one channel. The number of confirm messages in D-CAT is denoted by $x(x \cdot \ X)$.

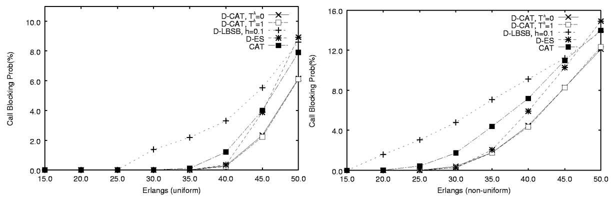
Table I shows the total number of messages transmitted between the cells and the delay of message transmission for importing X channels for CAT, D-LBSB, D-ES, and D-CAT. The delay for message transmission for channel acquisition in D-CAT leads to $2\delta + \delta_d$ so long as one imported channel has no need to be confirmed, i.e., $\theta = 0$, and $4\delta + \delta_d$ only if all of the imported channels need to be confirmed, i.e., $\theta = 1$.

It is observed that D-CAT yields the least message complexity compared with all of other algorithms. Since the channels in D-CAT are held by each cell, there is no need to perform the channel returning operation as in D-ES and D-LBSB. Furthermore, an importer in D-CAT confirms the reserved channels only once since it has the information of all the reserved channels at each exporter. D-ES, on the other hand, has to check every reserved channel it attempts to import sequentially and in the worst case it needs to confirm multiple times for importing a single channel. Furthermore, since D-CAT can import multiple channels within one channel allocation operation and the reserved channels are treated with low priorities, the message delay in most cases in D-CAT will be reduced to $2\delta + \delta_d$.

TABLE I
Implementation cost comparison of the algorithms.

Scheme	Number of messages	Message delay
CAT	$N + 4X + 1$	2δ
D-LBSB	$2(N-1) + 2(\|N_i\| + 3)X$	$4\delta(1 + 3X)$
D-ES	$(3\|N_i\| + 3n_p m + n_u)X$	$(2(1+m)\delta + \delta_d)X$
D-CAT	$3\|N_i\| + x$	$2(1+\theta)\delta + \delta_d$

B. Simulation Experiments

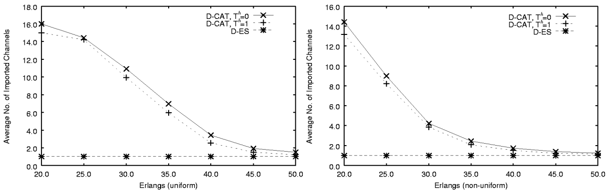


(a) (b)

Fig. 3. Comparison of call blocking probability.

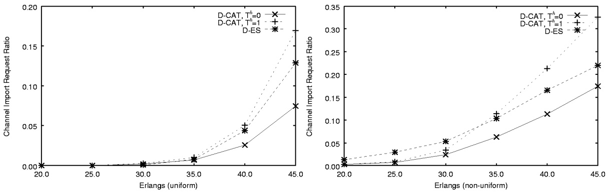
We conducted simulation studies to evaluate D-CAT and compare it with D-LBSB, D-ES, and CAT in terms of call blocking probability with uniform and non-uniform traffic. The results shown in Figures 3–5 were obtained with 90% confidence interval and within 5% of the sample mean. The simulated cellular system contains $N = 15 \times 15$ hexagonal cells as shown in Figure 1. Each cell is initially assigned $S_c = 40$ channels. Incoming call arrival at each cell is assumed to follow a Poisson process with a mean λ. The holding time of a call is assumed to be distributed based on an exponential distribution with a mean $1/\mu$ of 180 secs (3 mins). The degree of coldness at a cell, h, in D-LBSB is 0.1. The parameters used in the simulation for CAT and D-LBSB are chosen from the best combination to yield low call blocking probability. The number of reserved channels at each cell in both D-ES and D-CAT is 1. The target threshold, T_i^t, in D-CAT is determined by using Equation 1. The heavy threshold, T^h, was examined in the simulation study for the cases of $T^h = 0, 1$. Two kinds of call demands, uniform and non-uniform, were simulated for the algorithms under consideration. In the previous case, call arrival at each cell is identical, whereas in the latter case, a cell can get congested from time to time. That is, a cell gets congested from λ to 3λ with a probability of 0.001 and a congested cell will return to the normal state with a probability of 0.01.

Figures 3(a) and 3(b) show the call blocking probability of all the algorithms under consideration. It is observed that



(a) (b)

Fig. 4. Number of channels within one importation operation.



(a) (b)

Fig. 5. Channel import request ratio.

D-CAT outperforms others in terms of the call blocking probability. In D-LBSB and CAT, the co-channel locking scheme is over conservative where some channel locking is not necessary in order to avoid co-channel interference, and the channel lender selection scheme is over pessimistic where a heavy cell is not allowed to borrow any channels from moderate cells. Even though D-CAT and D-ES behave similarly when the call demand is below 40 Erlangs (i.e., the cell capacity), D-ES degrades faster than D-CAT. This is because that D-CAT employs a channel selection scheme that always selects the best channels to import from the channel candidates and assigns the best channel to the incoming calls. It is also observed that the value of the heavy threshold, T^h, has no significant impact on the call blocking probability.

In order to evaluate the efficiency of D-CAT, the number of free channels that can be imported during one channel acquisition opertion and the frequency of requests for free channels were examined in comparision with D-ES. Figures 4(a) and 4(b) show the average number of channels successfully imported during one channel acquisition operation in D-CAT. The more the number of channels imported in one channel acquisition operation, the lower the overhead cost needed for transmitting the messages between the cells and for running the channel allocation algorithm. It is observed that D-CAT can find out multiple free channels for a heavy cell each time under most practical operation conditions, e.g., on an average more than 10 channels in a call demand of 30 Erlangs and around 3 channels even in a call demand near to 40 Erlangs as shown in Figure 4(a).

Figures 5(a) and 5(b) show the channel import request ratio, defined by the ratio of the number of channel import requests to the total number of call arrivals, in D-CAT and D-ES. It is observed from Figures 5(a) and 5(b) that T^h has little effect on the channel import request ratio in D-CAT when the call demand is low, but the effect becomes greater afterwards. This result indicates that if the calling traffic is not very high, i.e., lower than the cell capacity, $T^h = 1$ is preferable. On the other hand, when the calling traffic becomes near to or greater than the cell capacity, letting $T^h = 0$ reduces the channel import request ratio, resulting in a much lower overhead for running the algorithm. It is observed that in D-CAT when $T^h = 0$ the channel import request ratio is lower than that in D-ES for both uniform and non-uniform calling traffic. For example, the channel import request ratio in D-CAT is lower than that in D-ES by over 30% when the call demand is near to or greater than 40 Erlangs. This means that a cell in D-ES frequently needs free channels and has to run the channel acquisition algorithm more often than D-CAT.

V. CONCLUSIONS

In this paper, a distributed dynamic channel allocation algorithm called D-CAT based on a two-threshold scheme has been proposed for mobile cellular networks. It has been shown that D-CAT outperforms other centralized and distributed algorithms in terms of the call blocking probability. The D-CAT algorithm also yields a lower message complexity and a shorter message transmission delay than the existing distributed algorithms. It has been observed that a heavy cell in D-CAT can import multiple channels, in comparison with only one channel in D-ES, during each channel acquisition operation; e.g., a heavy cell can import around 3 channels on an average in D-CAT when the call demand is near to the cell capacity. Furthermore, the import request ratio in D-CAT is lower than that in D-ES by over 30% when the call demand is near to or greater than the cell capacity.

References

[1] Ed. J.D. Gibson, Mobile Communications Handbook, CRC Press, 1999.
[2] G. Cao and M. Singhal, "Efficient distributed channel allocation for mobile cellular networks," in Proc. IEEE 7th Int. Conf. Comput. and Commun. Networks, October 1998, pp. 50–57.
[3] S.K. Das, S.K.Sen, and R. Jayaram, "A dynamic load balancing strategy for channel assignment using selective borrowing in cellular mobile environment," Wireless Networks, vol. 3, no. 5, pp. 333–348, October 1997.
[4] X. Dong and T.H. Lai, "Distributed dynamic carrier allocation in mobile cellular networks: Search vs. Update," in Proc. IEEE 17th Int. Conf. Dist. Comput. Syst., 1997, pp. 108–115.
[5] N. Garg, M. Papatriantafilou, and P. Tsigas, "Distributed list coloring: How to dynamically allocate frequencies to mobile base stations," in Proc. 8th Annual IEEE Symp. Parallel and Dist. Process., October 1996, pp. 18–25.
[6] Y. Zhang and S.K. Das, "An efficient load-balancing algorithm based on a two-threshold cell selection scheme in mobile cellular networks," Comput. Commun., vol. 23, no. 5-6, pp. 452–461, March 2000.
[7] M. Zhang and T.S.P. Yum, "Comparisons of channel-assignment strategies in cellular mobile telephone systems," IEEE Trans. Vehi. Tech., vol. 38, no. 4, pp. 211–215, November 1989.
[8] A. Baiocchi, F.D. Priscoli, F. Grilli, and F. Sestini, "The geometric dynamic channel allocation as a practical strategy in mobile networks with bursty user mobility," IEEE Trans. Vehi. Tech., vol. 44, no. 1, pp. 14–23, February 1995.
[9] S.K. Das, S.K. Sen, R. Jayaram, and P. Agrawal, "An efficient distributed channel management algorithm for cellular mobile networks," in Proc. IEEE Int. Conf. Universal Personal Commun., October 1997, pp. 646–650.
[10] G. Cao and M. Singhal, "Distributed fault-tolerant channel allocation for mobile cellular networks," in Proc. INFOCOM'99, March 1999, pp. 584–591.
[11] G. Cao and M. Singhal, "An adaptive distributed channel allocation strategy for mobile cellular networks," J. Parallel and Dist. Comput., vol. 60, no. 4, pp. 451–473, April 2000.
[12] U. Black, Mobile and Wireless Networks, Prentice-Hall PTR, 1996.
[13] L. Lamport, "Time, clocks, and the ordering of events in a distributed system," Commun. of the ACM, vol. 21, no. 7, pp. 558–565, July 1978.