

Token based distributed dynamic channel allocation in wireless communication network

Abdul Quaiyum Ansari · Premchand Saxena ·
Koyel Datta Gupta

Received: 20 May 2013 / Accepted: 6 July 2014 / Published online: 1 August 2014
© CSI Publications 2014

Abstract Channel allocation in wireless communication network plays a crucial role in the performance of the network. Fixed scheme for channel allocation does not account for any non-uniform traffic in the network. Dynamic scheme, on the other hand, ensures that whenever a cell requires a channel, it is allocated pertaining to the frequency reuse constraints. The fact that distributed dynamic channel allocation technique has recently gained popularity is due to its high reliability and scalability. In this paper we propose a token based mutual exclusion technique to solve the problem of distributed dynamic channel allocation. We will show through simulation, how message exchange between cells is reduced in token based relaxed mutual exclusion as opposed to other relaxed mutual exclusion techniques.

Keywords Channel allocation · Mutual exclusion · Interference · Message complexity · Acquisition delay · Blocking probability

1 Introduction

One of the major issues in wireless network is limited availability of transmission spectrum which therefore must be shared by several users. The wireless users within each other's transmission range can cause significant mutual interference. To address the issue of spectrum reuse, the concept of cellular communications was introduced. The basic concept being that a given geographical area is divided into hexagons called cells. The available frequency spectrum is divided into non-overlapping channels where each channel is capable of supporting a communication session. The concept of cellular system allows different cells (that are at least separated by a minimum reuse distance [1]) to use the same channel simultaneously. The assignment of the channels to the cells are subjected to three different types of constraints; co-channel, adjacent and co-site interference. A channel allocation strategy should follow the principle of frequency reuse for better and effective communication. Many approaches have been used to resolve the problem of effective channel allocation. The major categories for assigning channel to cells are either fixed [2] or dynamic [3–9]. A combination of fixed and dynamic channel allocation has also been used termed as hybrid channel allocation [10–12]. A comprehensive evaluation of some of these techniques have been presented in [13–15]. Due to the non-uniform pattern of the traffic in the wireless network, fixed channel allocation scheme may not always provide the optimal solution. On the contrary, dynamic channel allocation attempts to alleviate the problem of non-uniform traffic in the network. For many years, centralized dynamic channel assigning scheme [10] was nurtured by researchers. In this case all available channels are maintained centrally (mobile switching station) and allocated to respective cells. But a failure in the

A. Q. Ansari
Department of Electrical Engineering, Jamia Millia Islamia,
New Delhi, India
e-mail: aqansari@ieee.org

P. Saxena
School of Computer & Systems Sciences, Jawaharlal Nehru
University, New Delhi, India
e-mail: premchand_saxena@yahoo.com

K. Datta Gupta (✉)
Research Scholar, Department of Electrical Engineering, Jamia
Millia Islamia, New Delhi, India
e-mail: koyel.dg@gmail.com

central system can lead to a catastrophic effect. Our paper focuses on distributed dynamic channel allocation (DCA) technique [16]. To reduce the co-channel interference and utilizing the available bandwidth in an efficient way, effective resource sharing is required. The problem can be seen as a special case of mutual exclusion. In this case no two interfering cells can use the same channel at the same time but the channel can be shared between non-interfering cells.

1.1 Related work

Over the years many algorithms have been proposed for dynamic channel allocations which handle the non-uniform traffic issues. The authors [1] propose a general algorithm that ensures relaxed mutual exclusion for a single resource (in this case a channel). The study ensures that a given critical resource (channel) might be used simultaneously at different sites provided they are not mutually interfering. Though the work lowers the channel acquiring delay but it enhances the message complexity as the number of call requests increases. A Multi-Agent System proposed in [4] addresses the resource allocation problem through negotiation. But the work does not consider users mobility and their precedence. The authors [9] present an auction mechanism where each user(node) can bid for a channel and the second highest bidder wins. The authors [10] presents a hybrid channel allocation algorithm in which the base station sends a multi-level hot-spot notification to the Mobile Switching Station on each channel request that cannot be allotted locally at the base station. The Channel acquiring time is increased when channels are not available with the base station and the mode of channel allocation is centralized. The study [11] presents a hybrid channel assignment scheme where each cell is in charge of a group of local channels and shares a single pool of global channels with a traffic prediction module to calculate the possible number of local channels a cell will need in the subsequent time periods, taking into account recent traffic history of the cells in addition to their daily traffic periodicities. The algorithm uses centralized dynamic allocation which can prove to be unreliable if there is a problem in the global pool of channels. Another work [17] on distributed dynamic channel allocation (DDCA) algorithm proposed that heavily loaded regions acquiring a large number of communication channels, while their lightly loaded neighbours get assigned fewer channels. The algorithm distributed the responsibility for channel allocation among the mobile service stations of the network. The algorithm keeps the number of handoffs to a minimum as it did not induce any intra-cell handoffs. The proposed scheme has

a low overhead but has higher message complexity. The paper [18] proposes a distributed channel acquisition algorithm by integrating the channel search and update algorithm into the channel acquisition algorithm. By retaining the borrowed channels, the algorithm adapts to the varied network traffic.

All the earlier techniques and work discussed above have considerably high message complexity resulting in greater channel allocation time. The message complexity is low and almost constant in the proposed technique reducing the overall channel acquisition delay.

In this paper we present a modified token based mutual exclusion technique which solves the channel assignment problem. This paper shows the token based mutual exclusion algorithm, if used under some special consideration, can perform better than message passing counterparts. The conflicts arising due to two cells requesting for the same channel is solved with a Fuzzy rule-based system. The Fuzzy rule-based system categorizes the requests as “high”, “medium” and “low” priority.

The paper is organized as follows: Preliminaries are presented in Sect. 2. Section 3 presents problem formulation. Section 4 gives an overview about the performance of the proposed algorithm. The work is concluded in Sect. 5.

2 Preliminaries

The distributed mutual exclusion algorithms have been categorized as permission based, token based and quorum based.

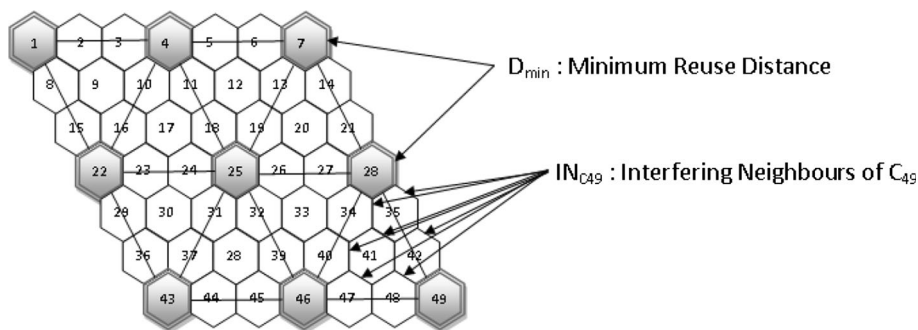
2.1 Permission based algorithms

In these algorithms [1, 17–20], each process (nodes in case of channel allocation problem) maintains a list of requests. Whenever a process tries to use a shared resource (channel) it sends request to all processes. The processes receiving the request give permission if it is not using the resource or does not intend to use the resource. The process generating the request can only use the resource once it receives permission from all the other processes.

2.2 Token based algorithms

In these algorithms a process possesses a token. The presence of the exclusive token ensures mutual exclusion. The possession of a system-wide unique token gives a process the right to access a shared resource in mutual exclusion [21].

Fig. 1 A 7×7 cellular system



2.3 Quorum based algorithms

Each site requests for permission to capture the critical resource from a subset of sites (called a quorum). Any two quorums contain a common site. This common site is responsible to make sure that only one request possesses the critical resource at any time [22].

Since token based mutual exclusion algorithm has not been used in channel allocation and shows promise in this field we have proposed a distributed dynamic channel allocation technique using the concept of token based mutual exclusion.

3 Problem formulation

In this section, we formally describe in details the system model and the formulation of the channel allocation problem.

3.1 System model

A distributed algorithm is presented that realizes mutual exclusion among N users in a cellular network. We are going to develop a model which is token-based and use multiple tokens to support the concept of reusability of channels among non-interfering cells. We consider a wireless communication system with 2D structure of 7×7 cells (7 rows and 7 columns) with each cell having 6 neighbours as shown in Fig. 1. It has been assumed that there are 410 channels in the system. The minimum reuse distance D_{min} is considered as 3. At present uniform distribution of traffic load has been considered. The presence of thermal noise and power interference from neighbouring cells may actually reduce the number of available channels to approximately 400.

In this system we use multiple tokens instead of 1 since there are multiple channels (which acts as the critical resource) which can be simultaneously acquired (in a non-

interfering way) by all the cells. The channels are represented by an individual token T_k , $k \in$ all channels.

- C_i Cell with number i
- T_{C_i} Stores the list of tokens currently being used by cell C_i
- S_i Maintains the status of Token T_i (locked or unlocked)
- $Temp_{C_i}$ Temporary list of tokens hold by C_i and IN_{C_i}
- L_{C_i} Lamport's logical clock for each cell C_i
- T Array of tokens

A single token can be acquired by multiple cells provided they do not cause any co-channel interference. The term co-channel interference means crosstalk from two different radio transmitters using the same frequency. The co-channel interference can be reduced when same channel is assigned to two different cells simultaneously if and only if the said cells are separated by minimum reuse distance. Reuse distance is defined as the distance between the centers of the cells that use the same channel. Each cell maintains a list of tokens being used currently. To support a newly originated call or a handoff call (an ongoing call travelling from one cell to other) the cell requests for a new channel. In order to procure a channel when a cell tries to acquire a new token it checks the list of tokens used by the interfering neighbours. A conflict occurs if two cells which can cause co-channel interference to one another tries to grab the same token simultaneously. In this case the one with higher priority gets the channel. The priority is set based on a set of fuzzy rules. A channel failure may occur due to synchronization problem. The channel failure should be resolved immediately by allocating another available channel on a high priority basis. The problem of adjacent interference is reduced by using substantial band between two successive spectrums.

3.2 The algorithm

- Input: Call requests
- Output: Calls permitted and blocked

3.2.1 Data structure

```
StructToken      /* Structure of token representing the channels*/
{
  int Req[49]; /*Req is the request to grab the token*/
  bool S;      /*Status of the token locked or unlocked*/
} T[400]
Struct Cell      /*Structure of Cell containing list of interfering neighbours */
{ /*and list of tokens being used currently*/
  int INC[18];
  int Tc[400];
} C[49]
```

3.2.2 Initialization

/*All token status and requests are initialized to 0 indicating not in use*/

/*The Token list of each cell is empty*/

```
1. For m = 0:48
   For k = 0:399
     C[m]. Tc[k] = -1 // C[m] denotes Cm i.e. mth numbered Cell
   End For
 End For
 For k = 0:399
   For i = 0:48
     T[k].Req[i] ← 0
   End For
   T[k].S ← false
 End For
 TempCi ← null
 Count ← 0
```

3.2.3 Computing the interfering neighbours

In the proposed algorithm, when a cell C_i requests for a channel, the interfering neighbours of C_i is computed. The distance between two cells is defined as the Euclidean distance between the centres of the two cells. The distance between two cells $C_1(x_1, y_1)$ and $C_2(x_2, y_2)$ is :

$$\text{dist}(C_1, C_2) = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2} \quad (1)$$

A number D_{\min} specifies the minimum reuse distance. A channel can be shared by various cells provided that the distance between each pair of cell is equal or greater than

D_{\min} . Thus each cell C_i has a set of interfering neighbours IN_{C_i} which is a set of cells whose distance to C_i is smaller than D_{\min} .

$$IN_{C_i} = \{ C_j; \text{dist}(C_i, C_j) < D_{\min}, i \neq j \} \quad (2)$$

3.2.4 Call request-searching and acquiring a token

A call request is raised by a cell C_i . The Cell looks for the most-recent token usage information before initiating a search. The list of tokens already in use by the interfering neighbours is noted and a free token is identified. The tokens used by C_i should also be eliminated to avoid co-site channel interference. The token T_k is locked by C_i and the token list used by C_i is updated by adding T_k to the list. As soon as C_i finishes the use of T_k , T_k is released and removed from the token usage list of C_i . If at any point when C_i makes a request and no free token is available, the request of C_i is dropped.

1. A cell $C[m]$ sends a request for a channel to its Base Station.
2. For $i = 0; \text{sizeof}(C[j].T_c) \forall C[j] \parallel C[j] \in C[m].IN_C \ \& \ j = 0:48, 0 \leq m \leq 48$

```
TempCi ← C[j].Tc
End For
```
3. For $k = 0:399$

```
For h = 0: sizeof(TempCi)
  If ((T[k] ≠ TempCi[h]) && (T[k].S <> true))
    T[k].Req[m] ← LCm
  For x = 0:48
    If(T[k].Req[x] ≠ 0)
      Count++
    If(Count == 1) /* Only one cell is trying to grab the Token*/
      T[k].S ← true;
      C[m]. Tc[k] ← -1 // C[m].Tc = C[m].Tc + T[k]
      T[k].S ← false ;
      T[k].Req[m] ← 0
      TempCi ← null
    End If
  End For
End For
```
4. If no free tokens are available then cell $C[m]$ drops the call request.

3.2.5 Resolving conflicts

If there is a situation when two interfering cells try to grab the same token, the cell which raised the first request (based on logical time stamping) will lock and use the first free token. The other cell will be allocated the next available token if the

cells are interfering. When the logical time stamp of two conflicting cells does not differ, fuzzy rules are used to set the priority of the request in each cell. A request with higher priority is allowed to grab the said token.

- (1) If two cells try to grab a token at the same time then the logical timestamp L_{C_i} and L_{C_j} is checked to identify the cell which has raised the request first. That cell gets the chance to grab and lock the first available free token.

If $L_{C_i} < L_{C_j}$, C_i blocks the token and C_j continues the search.

- (2) If two cells try to grab a token at the same time and their logical timestamps L_{C_i} and L_{C_j} are identical, the following Fuzzy rules are applied to set the priority and identify the cell which will lock the said token.

If ($L_{C_i} == L_{C_j}$)

- (i) If the call request in any cell is made to emergency number the priority of request is set to high.
- (ii) If the request is for handoff the priority of request is set to medium.
- (iii) If the request is for new call to trivial number the priority of request is set to low.

$\mu_c = \text{Type of Call/Emergency Call} \vee 0.5 \times \text{Type of Call/Hand-off/}$ If the call is to an emergency number $\mu_c \approx 1$ and if the call is ordinary $\mu_c \approx 0$.

The priority of the request determines which cell will grab the token.

3.2.6 Terminating a call-releasing a token

- (1) When the call terminates

$$C[m].T_c[k] \leftarrow -1 // C[m].T_c = C[m].T_c - T[k]$$

3.2.7 Failure of a channel

If at any instant of time a single channel fails, the call using the channel is handed over to some new channel (token). The channel(s) which failed is temporary removed from the list before a channel acquisition request is serviced. This might lead to higher blocking probability.

3.2.8 Algorithm overview

Lemma 1 Let T be the communication time between mobile host and Base Station, then the message complexity of the channel allocation technique is $2 \times T$.

Proof In the first step of the proposed technique the mobile host (at the arrival of a call request) requests for a channel from the Base station. The time taken for this is T. The Base station uses the recent token usage history and identifies a free token which is not used by an interfering neighbour. At this point of time the base station does not need to communicate with other any other base station.

The base communicates back to the mobile host in time T. The total message complexity of the technique is $2 \times T$.

Once the base station identifies and allocates a channel it updates the token usage history.

Lemma 2 Let T be the communication time between mobile host and Base Station, then the Channel acquisition time of the channel allocation technique for uniform and non-uniform traffic is $2 \times T$.

Proof Let us consider non-uniform load distribution throughout the network.

Then the channel acquisition delay can be calculated as $2 \times T(\text{RTT between mobile host and base station}) + dl(\text{latency involved in searching non-interfering channels})$

For $D_{\min} = 3$:

$$\begin{aligned} \text{The number of interfering neighbours for Cell } C_i &= \sum_i IN_{C_i} = \{ C_j : \text{dist}(C_i, C_j) < D_{\min}, i \neq j; i, j = 0 : 48 \} \\ &= \text{The number of boundary edges} \times D_{\min} = 18 \end{aligned}$$

If content addressable memory is used for the token list the latency is reduced to maximum number of token used by any interfering cells.

Hence the lists of tokens used by IN_{C_i} are searched simultaneously.

Let $t_j =$ number of tokens used by C_j where $C_j \in IN_{C_i}$
Therefore $dl = \max(t_j) \times \text{search time of each token}$

It is assumed that let $T = 1$ ms and each search for token entry requires less than 0.75 ns.

If content addressable memory is used for the token list the latency is reduced to maximum number of token used by any interfering cells.

For uniform traffic:

$$\begin{aligned} \text{Maximum number of tokens possessed by any cell} &= \text{The maximum no. of alive calls in the cell} \\ &= \lfloor (\text{No. of channels} / \text{No. of interfering neighbours}) \rfloor \\ &\approx 21 \end{aligned}$$

Now $dl = 21 \times .75$ (in ns) therefore, $2 \times T$ (in ms) \gg Searching Time for free tokens (in ns). hence, the channel acquisition delay is reduced to $2 \times T$ for uniform traffic.

For non-uniform traffic: the number of tokens to be searched by any cell $C_i =$ number of tokens possessed by

the interfering + number of tokens possessed by the cell itself < 400. Since, the total number of tokens is 400.

Now, for any highly loaded cell C_j (with maximum number of alive calls and $C_j \in IN_{C_i}$)—let the number of tokens used be $a \times 400$ where a is the fraction of total tokens and $0 \leq a < 1$ (Since $a = 1$ implies no free token available.)

Since the tokens used by all interfering neighbors are searched simultaneously

$$dl(\text{searching time delay}) = (0.75 \times a \times 400)ns = (300 \times a) ns < 3 \times 10^{-1} ms \text{ (since } a < 1)$$

Therefore, $2 \times T$ (in ms) > dl.

For practical scenario ‘a’ should be less than 0.6.

4 Simulation

In Sect. 3, we described the system model, and subsequently presented a token based distributed channel allocation algorithm for the system. We have used NS-2 for the simulation. The wireless model supports simulations of wireless cellular network. For simulation the entire topology is divided into 7×7 grid each of 70 sqm. The algorithm works for non-uniform traffic. The propagation delay of the wireless link is set to 1 ms and for the wired link (between Base Station and MSS) has a delay of 10 ms. The message complexity of the algorithm is low making it energy efficient. To review the performance of a resource allocation technique, two other vital factors are the delay between request generated and actual allocation and the number of call blocks and call drops. The proposed channel allocation scheme is evaluated and compared with some existing techniques in terms of these three different criteria and plotted using xgraph utility with NS-2.

- (i) Message Complexity
- (ii) Channel Acquisition Delay
- (iii) Call Blocking Rate

Message complexity is the number of messages exchanged at the time of acquiring a channel.

On the other hand channel acquisition delay defines the total time taken to acquire a channel after a call request is raised. The call blocking probability refers to the number of channels allocated to the actual number of requests generated.

For analysing the above mentioned criteria we assumed a wireless communication system with 2D structure of 7×7 cells (the entire topology spread is restricted to 490×490 m # $\$$ topo load_flatgrid 490 490 70) with each cell having 6 neighbours as shown in Fig. 1. It has been assumed there are 410 channels in the system. The propagation delay of the wireless link is set to 1 ms. The rate of call arrival is assumed to be the Poisson distribution and the life time of a call is assumed to be 180 s on an average. The distance between any two cells is calculated using the Euclidean distance between the centres of the two cells and $D_{min} = 3$. Hence, the number of interfering neighbours is 18. The presence of thermal noise and power interference may reduce the actual number of channels to 400 (approximately). The xgraph utility is used to plot the simulation results. The message complexity, channel acquisition delay and call blocking rate is considered as follows:

- (i) Message Complexity: The message complexity is measured in terms of the communication time between the mobile host and the Base Station. Let the communication latency on each direction be measured as T , then the total time for all call requests (irrespective of the no. of calls in the system) is $2 \times T$ and the number of messages exchanged is 2. The message complexity as described in [18] is $2 \times m$ (in time T ; where m is the number of interfering neighbours i.e. 18 for this case). The messages exchanged for channel acquisition [1] (TSDP-Mercedes) is $(6 \times n + 9) \times d$ (in time T ; $d = \lfloor D_{min} \rfloor$ and $0 \leq n \leq 1$; $n + 1$ is the number of

Fig. 2 Simulation result of proposed algorithm showing message complexity for non-uniform traffic

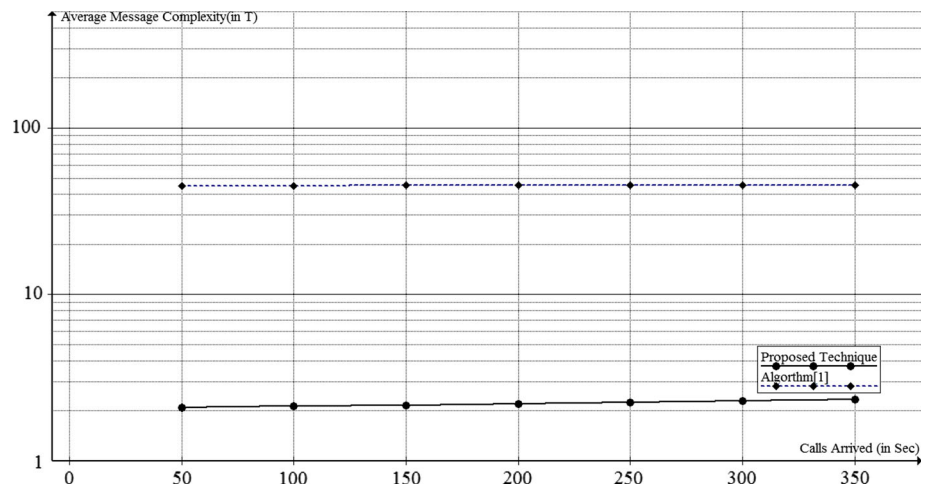


Fig. 3 Simulation result of proposed algorithm showing channel acquisition delay for non-uniform traffic

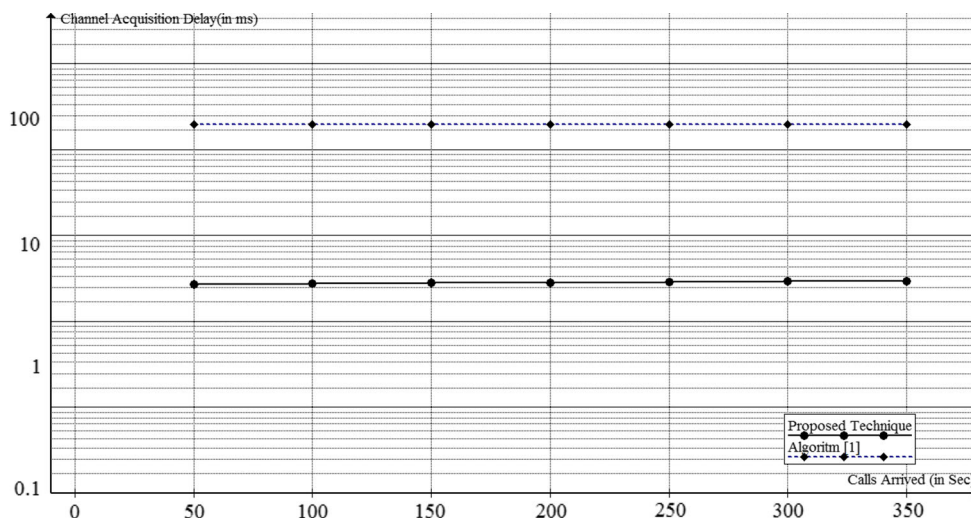
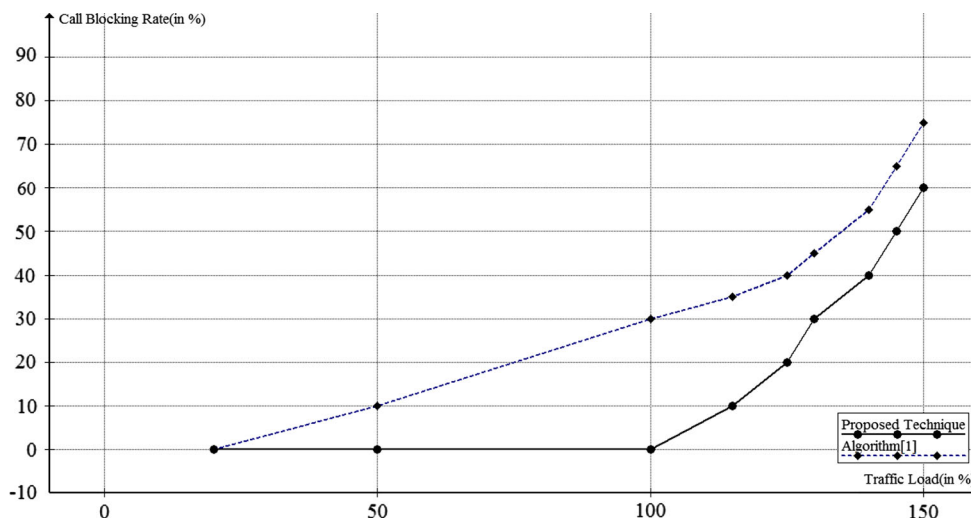


Fig. 4 Simulation result of proposed algorithm showing Call Blocking Rate for non-uniform traffic



searches; assuming second search is successful) which is more than the proposed algorithm (shown in Fig. 2).

- (ii) *Channel Acquisition Delay*: Considering both uniform and non-uniform load distribution in the network the channel acquisition delay can be calculated as $2 \times T$ (T is the communication time between mobile host and Base Station). The channel acquisition delay as shown in Fig. 3, when compared with [1] was found to be much less since the latency involved was mainly due to the message transmission between mobile service station and base station (Considering memory searching time to be insignificant) whereas the algorithm [1] (TSDP-Mercedes) has a delay of $2 \times (n + 1) \times TT$ where $0 \leq n \leq 1$; $n + 1$ is number of searches; TT = Communication time between two cells + Delay to search and generate appropriate reply for a channel acquisition

request (with the assumption that the second search will be successful).

- (iii) *Call Blocking Rate*: Assuming uniform distribution of traffic and equal availability of tokens for all cells, the blocking rate is insignificant till 1000 calls (assuming equal load distribution among all cells with $D_{min} = 3$). The blocking rate appreciates as the number of calls shoot over 2,000.

The call blocking rate of our algorithm (for non-uniform traffic) when evaluated against [1] performed better (shown in Fig. 4).

5 Conclusion

The paper proposed an approach of token based distributed mutual exclusion for solving the channel allocation

problem. The simulation results show that without downgrading the channel acquisition delay the message complexity has largely been reduced. The message complexity plays an important role since a large overhead can be reduced if no. of messages exchanged per channel request can be reduced. The use of Fuzzy rule ensures that a hand off call is always allocated a channel provided the cell requesting has provision for acquiring new channel without causing any interference. The system requires basic information structure and use of content addressable memory for searching tokens to reduce the channel acquisition delay. The algorithm has been tested with non-uniform load distribution and the performance did not show any significant degradation. The concept of load balancing can be considered for future work.

References

- Jiang J, Lai TH, Soundarajan N (2002) On distributed dynamic channel allocation in mobile cellular networks. *IEEE Trans Parallel Distrib Syst* 13(10):1024–1036
- Yue W, Miyazaki K, Deng X (2004) Optimal channel assignment in wireless communication networks with distance and frequency interferences. *Compu Commun (Elsevier)* 27(16):1661–1669
- Kaabi F, Ghannay S, Filali F (2010) Channel allocation and routing in wireless mesh networks: a survey and qualitative comparison between schemes. *Int J Wirel Mob Netw* 2(1):132–150
- Kamble M, Gupta R (2010) DDCA in cellular network using fuzzy rule based multiagent system. *Int J Compu Appl* 1(17):81–84
- Ghaderi M, Sridharan A, Zang H, Towsley D, Cruz R (2009) TCP-aware channel allocation in CDMA networks. *IEEE Trans Mobil Comput* 8(1):14–28
- Hung Kuo-Ming, Wang Yao-Tien, Hsieh Ching-Tang (2009) Hierarchical genetic algorithm for dynamic channel units allocation in TDCDMA/TDD system. *Int J Wirel Mobil Netw* 1(2):103–116
- Kamaluddin Khaja (2009) Dynamic channel allocation in mobile wireless cellular networks in heavy load conditions–time slot division method. *Int Conf Commun Compu Power* 410–412
- An J, Hines EL, Leeson MS, Sun V, Wren, and Iliescu DD (2007) Genetic algorithms and Fuzzy logic for dynamic channel allocation in cellular radio. *Radio Wirel Symp IEEE* 19–22
- Sun J, Modiano E, Zheng L (2006) Wireless channel allocation using an auction algorithm. *IEEE J Sel Areas Commun* 24(5):1085–1096
- Ahmed RanaEjaz (2010) A channel allocation algorithm for hot-spot cells in wireless networks. *J Adv Inf Technol* 1(3):136–140
- Salah-Eldeen S, Elnahas A, Ghoniemy S (2007) A hybrid distributed channel allocation. *Conf Inf Commun Technol IEEE* 2007.115–121
- Vidyarthi Geetali, Alioune Ngom, Ivan Stojmenović (2005) A Hybrid Channel Assignment Approach Using an Efficient Evolutionary Strategy in Wireless Mobile Networks. *IEEE Trans Veh Technol* 54(5):1887–1895
- Mishra MP, Saxena PC (2012) Survey of channel allocation algorithms research for cellular systems. *Int J Netw Commun* 2(5):75–104
- Du W, Jia W, Wang G, Lu W (2009) Analysis of channel allocation scheme for wireless cellular networks. *Int J Ad Hoc Ubiquitous Comput* 4(3):201–209
- Papazoglou PM, Karras DA, Papademetriou RC (2007) A critical overview on the recent advances in channel allocation strategies for voice and multimedia services in wireless communication systems and the applicability of computational intelligence techniques. *Math Methods Comput Tech Non-Linear Sys Intell Syst* 314–320
- Cimini LJ, Foschini GJ (1993) Distributed dynamic channel allocation algorithms for microcellular systems. *Wirel Commun* 217: 219–241. *Kluwer International Series in Engineering and Computer Science*
- Prakash Ravi, Niranjana G Shivaratri, Mukesh Singhal (1999) Distributed dynamic fault-tolerant channel allocation for cellular networks. *IEEE Trans Veh Technol* 48(6):1874–1888
- Cao G, Singhal M (2000) An adaptive distributed channel allocation strategy for mobile cellular networks. *J Parallel Distrib Comput* 60:36–42
- Cao G, Singhal M (2000) Distributed fault-tolerant channel allocation for cellular networks. *IEEE J Sel Areas Commun* 18(7):1326–1337
- Ricart G, Agrawal AK (1981) An optimal algorithm for mutual exclusion in computer networks. *Comm. ACM* 24(1):9–17
- Suzuki I, Kasami T (1985) A distributed mutual exclusion algorithm. *ACM Trans Comput Syst* 3(4):334–349
- Juneja R, Kumar V (2012) Quorum based distributed mutual exclusion algorithms in mobile networks. *Int J Comput Appl* 38(3):22–25