A Distributed Artificial Intelligent Scheme for Channels Allocation in Cellular Communication Networks

أسلوب ذكاء صناعي موزع لتنظيم القنوات في شبكات الاتصال الخلوية

By

Arafat A. Abu-Mallouh

(20033240159)

Supervisor

Dr. Hussein Al-Bahadili

This thesis is submitted to the Department of Computer Science, Graduate College of Computing Studies, Amman Arab University for Graduate Studies in fulfilment of the requirements for the degree of Master of Science in Computer Science.

Graduate College of Computing Studies
Amman Arab University for Graduate Studies
(May, 2008)

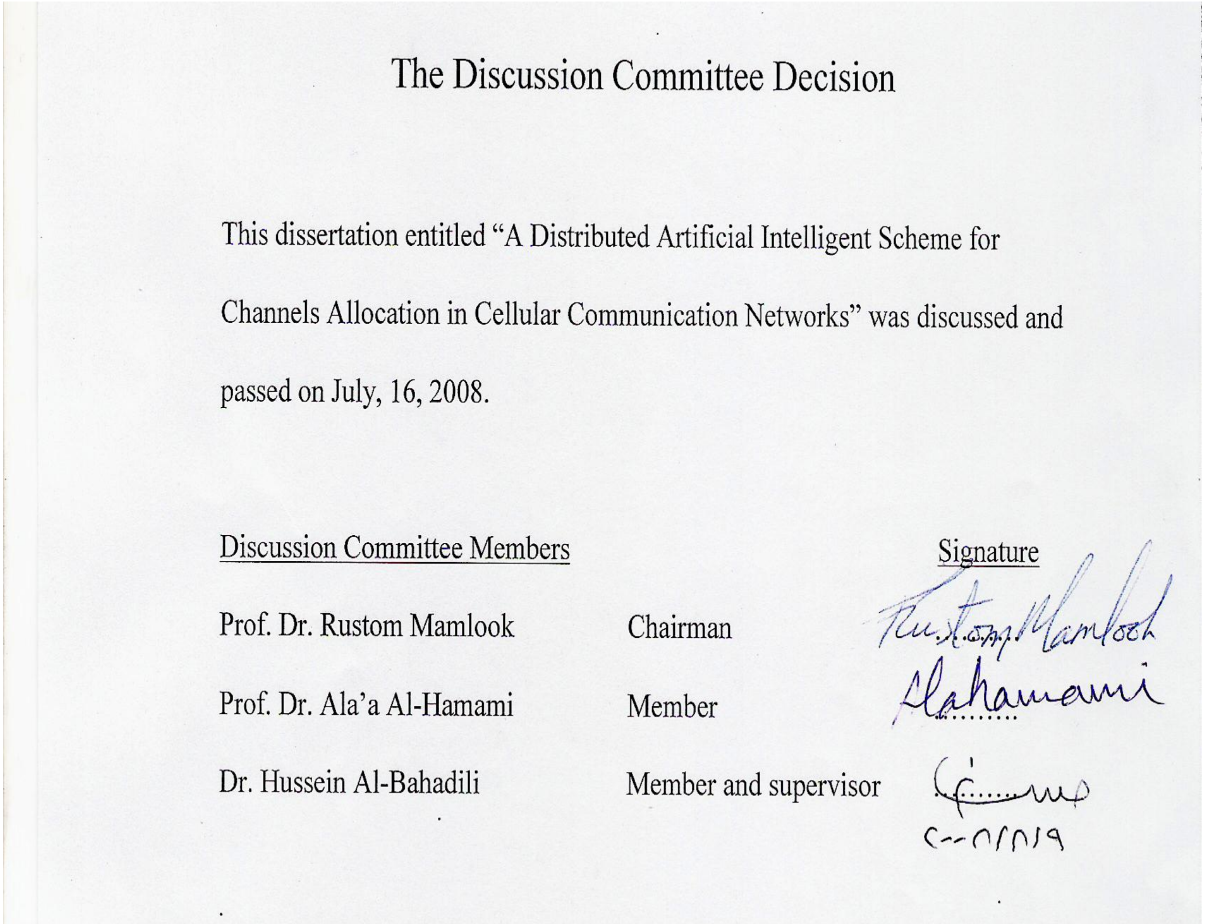
Authorization

I, Arafat A. Abu Mallouh, authorize Amman Arab University for Graduate Studies to supply copies of my dissertation to libraries, institutions, organizations or persons when required.

Name: Arafat A. Abu Mallouh.

Signature:

Date: 16/8/2008



The Discussion Committee Decision

This dissertation entitled “A Distributed Artificial Intelligent Scheme for Channels Allocation in Cellular Communication Networks” was discussed and passed on July, 16, 2008.

Discussion Committee Members		Signature
Prof. Dr. Rustom Mamlook	Chairman	Rustom Mamlook
Prof. Dr. Ala’a Al-Hamami	Member	Alahamami
Dr. Hussein Al-Bahadili	Member and supervisor	٢٠٠٨/٨/١٩

Abstract

The main objective of this work is to develop and evaluate the performance of an efficient and a reliable scheme that can be used for Dynamic Channel Allocation (DCA) in Cellular Communication Networks (CCNs). Therefore, it is referred to as the DCA scheme. The scheme utilizes a well-known Distributed Artificial Intelligence (DAI) algorithm, namely, the Asynchronous Weak-Commitment (AWC) algorithm, in which a complete solution is established by extensive communication among a group of neighbouring collaborative cells forming a pattern, where each cell in the pattern uses a unique set of frequencies.

In order to minimize communication overhead among collaborative cells, thus enhancing the performance of the DCA scheme, a token-based mechanism is introduced. In the DCA Scheme, each pattern has a token that circulates among the cells to achieve two objectives: passing channels that are currently allocated for each cell, and only allow a cell holding the token to update its resources. Since the token is continuously circulating among collaborative cells; a cell updates its resources if it is equal to or less than a certain threshold value. Otherwise, it just bypasses the token to the next cell to minimize delay.

An efficient discrete event simulator is developed using Borland Java Builder9 Enterprise edition. Despite its extreme capability and flexibility, the simulator is used to solve two channel allocation GSM scenarios that illustrate different network operation environments. The first scenario simulates a CCN in a suburban or a rural area that is characterized by a uniform initial load of 8 and 10 channels/cell, and relatively low and uniform traffic load of 0.2 calls/sec all over the network. The second scenario simulates a CCN in an urban downtown area, where the traffic load is centralized as in shopping or business districts. It simulates a "hot-spot" cell, in which the traffic load is varied from 0.4 to 1.0 calls/sec in step of 0.2. Each cell around the hop-spot cell has a traffic load of 0.4 calls/sec, while all other cells have a traffic load of 0.2 calls/sec. In this scenario, for the same traffic loads, the simulation repeated for different uniform initial loads of 4, 6, 8, and 10 channels/cell.

Simulations results show the effectiveness of the DCA scheme for solving channel allocation problems to achieve optimum bandwidth utilization and QoS level as it achieved average allocation efficiencies of over 95% for the first scenario, and over 85% for the second scenario.

Arabic Summary

ملخص

إن الهدف الرئيس لهذا العمل هو تطوير نظام كفؤ يمكن الإعتماد عليه و تقييم أدائه من أجل إستخدامه بطريقه ديناميكية لتنظيم القنوات في شبكات الإتصال الخليوية، لذلك تم تسمية هذا النظام الجديد "آلية لتنظيم القنوات بطريقة ديناميكية" (DCA scheme). إن (DCA scheme) يستخدم خوارزمية معروفة في مجال الذكاء الصناعي الموزع وهي خوارزمية الإلتزام الضعيف غير المتزامن (Asynchronous Weak-Commitment Algorithm)، وهذه الخوارزمية تقوم بإيجاد حل كامل للمشكلة من خلال الإتصال الواسع والشامل بين مجموعة من الخلايا المتجاورة والمتعاونة والتي تشكل نموذجاً، بحيث أن كل خلية في هذا النموذج تستخدم مجموعة منفردة من الترددات.

ومن أجل تقليل عبء الإتصال بين مجموعة من الخلايا المتعاونة وبالتالي تحسين أداء (DCA scheme)، تم تقديم آلية جديدة بالإعتماد على التبادل الحلقي (Token-Based). حيث أنه في (DCA scheme) كل نموذج من الخلايا يمتلك آلية تبادل حلقي تدور بين خلايا النموذج لتحقيق هدفين: أولاً تمرير معلومات عن القنوات المحجوزة حالياً في كل خلية، ثانياً السماح فقط للخلية التي تمسك أل (Token) بتحديث مصادرها من القنوات، حيث أن أل (Token) يدور بشكل مستمر بين الخلايا المتعاونة في النموذج الواحد، وكل خلية تحدث مصادرها من القنوات إذا كانت قيمتها أقل أو تساوي قيمة حدية معينة (Threshold_Value)، أما إذا كانت غير ذلك فإنه يتم تمرير أل (Token) للخلية التالية في النموذج لتقليل ألتأخير.

لقد تم تطوير محاكي حدثي منفصل بإستخدام لغة البرمجة (Borland Java) أستخدم لحل حالتين فقط من حجز القنوات في نظام أل (GSM) والتي توضح بيئات تشغيلية مختلفة للشبكة. الحالة الأولى تحاكي شبكة إتصال خليوية في ضاحية أو منطقة غير مكتظة والتي تتميز بحمل مبدأي متماثل و حمل مروري متماثل

قليل نسبياً في جميع أنحاء الشبكة. أما الحالة الثانية فهي تحاكي شبكة إتصال خليوية في مركز المدينة حيث الحمل المروري متركز في خلية معينة (Hot-Spot Cell)، وهذا الحمل المروري متغير، و في هذه الحالة الثانية و بالنسبة لنفس الحمل المروري، فقد تم تكرار المحاكاة لقيم مختلفة من الحمل المتماثل المبدأي.

إن نتائج المحاكاة تظهر فعالية (DCA scheme) لحل مشاكل تنظيم القنوات لتحقيق الدرجة المثلى في إستخدام عرض الحزمة ومستوى نوعية الخدمة، حيث كانت قيمة معدل الفعالية أكثر من 95% في الحالة الأولى وأكثر من 85% في الحالة الثانية.

Acknowledgment

During the time I spent on working to finish this thesis, I couldn’t forget the people who played a main role in helping me. Those people were the candles who burned part of their life's to lighten the road in front of me to succeed.

The first one of those wonderful people is my supervisor Dr. Hussein Al Bahadili, he dedicated his own valuable time to help, encourage, motivate me, and work with me to bring about my thesis in the best content and form. I can say Dr. Hussein was one of the people who had inerasable effect on my life.

Secondly, my parents, I think no words in the whole world can give them the credit they deserve, they dedicated their time working day and night to do the best for me, and I appreciate their efforts to put me on the right path and to support me. I hope this thesis is a little gift to say thank you my dear parents. The last person but not least is my brother Jamal. He was the first to guide me toward my MSc, he was my permanent motivator who didn’t get bored of urging me to do the best and to work harder, thank you Jamal.

Table of Contents

List of Figures

List of Tables

Abbreviations

ABT	Asynchronous Backtracking Algorithm.
AI	Artificial Intelligence.
AMPS	Advanced Mobile Phone Services
AWC	Asynchronous Weak-Commitment Algorithm.
BS	Base Station.
BSC	Base Station Controller.
BTS	Base Transceiver Station.
CCN	Cellular Communication Network.
CDMA	Code Division Multiple Access.
CEPT	Conference of European Posts and Telecommunications
CSP	Constraint satisfaction problem.
DAI	Distributed Artificial Intelligence.
DCA	Dynamic Channel Allocation.
DCSP	Distributed constraint satisfaction problem.
E	Efficiency.
FDMA	Frequency division Multiple Access.
GSM	Global System for Mobile communications.
HCA	Hybrid Channel Allocation.
IMTS	Improved Mobile Telephone System.
ISDN	Integrated Services Digital Network.

MAS	Multi-Agent System.
MS	Mobile Station.
MSC	Mobile Switching Center.
PSTN	Public Switched Telephone Network.
QoS	Quality of Service.
SCA	Static Channel Allocation.
S_R	Success Ratio.
TACS	Total Access Communications System.
TDMA	Time Division Multiple Access.

Chapter 1
Introduction

1.1. Introduction

A wireless communication network is defined as network that consists of a number of power-constraint communication devices utilizing radio waves to communicate with each other in a limited coverage area [Gar 06, Kou 07, Leu 07]. This allows communications devices to move around within the radio coverage area and still be connected to the network. In addition to the limited coverage area, the radio spectrum is a scarce resource in wireless networks. Thus, the current wireless communication networks use a cellular architecture [An 07, Lee 07, Leu 07]; in which the geographical area is divided into several coverage areas called cells, and the radio spectrum is divided into a number of channels using frequency division multiple access (FDMA) technique [Tan 03, Yan 05, Yan 07]. However, these channels can be further divided using time division multiple access (TDMA) [Leu 07, Spy 07] and code division multiple access (CDMA) techniques [Leu 07, Nav 02].

In such Cellular Communication Networks (CCNs), a channel *c* can be used by a cell *i* without any interference (co-channel) if it is not concurrently used by any other cell at a limited distance from cell *i* (called minimum reuse distance). Furthermore, for better quality of service (QoS) and to minimize adjacent channels interference, channels adjacent to *c* should not be used within cell *i* or any of its first-hop neighbors. Therefore, it is essential to devise suitable solutions for allocating channels to cells so as to efficiently utilize the scarce resources, to eliminate channel interferences, and to satisfy any other network operation environment and demand. In addition, solutions should meet the main design objectives, such as: minimum channel acquisition time, minimum number of denied or failed calls, minimum control message complexity, minimum communication overheads, and minimum network interruption [Fel 06, Jia 08, Leu 07, Rob 00, Sar 00].

A number of techniques have been developed to combat impairments in rapidly varying radio channels and to obtain high spectral efficiencies in CCNs. Some of those are channel coding and interleaving, adaptive modulation, transmitter/receiver antenna diversity, spectrum spreading, and dynamic channel allocation (DCA) [Kos 01].

In general, channel allocation schemes in a mobile wireless environment (e.g., CCNs) can be either static or dynamic. Static allocation offers negligible channel acquisition time and zero message complexity and works well at a low system load; the performance steadily decreases as network traffic load increases since many calls are dropped; in case of even temporary hot spots many calls may be dropped by a heavily loaded base station (BS) even when there are enough idle channels in the interference region of that BS. On the other hand, dynamic schemes provide better utilization of the channels at higher traffic loads albeit at the cost of higher acquisition time and some additional control messages [Fel 06, Ras 07].

The purpose of this research is to develop a DCA scheme by utilizing a well-known distributed artificial intelligent (DAI) algorithm, namely, the asynchronous weak-commitment (AWC) algorithm [Fal 04, Yok 00].

It can be used within any cell (BS) to satisfy its own load independent of other cells (BSs) within its interference region, and meet all design objectives mentioned above. In order to minimize data communication and network interruption, a token-based scheme is developed, in which a token is circulating between any group of collaborative cells passing information about the channels that are allocated or in operation within each cell. In addition, the token controls the resource allocation process by only allowing the cell that hold the token to update its resources.

In this research, results are presented for a number of simulation studies that combine effects of initial and traffic loads on a number of network parameters, such as the average number of channels that are successfully allocated or denied within each cell and over all the networks. In addition, the algorithm channel allocation efficiency, which is defined as the ratio between the number of channels that are allocated to the total number of channels (allocated and failed), is computed for individual cells or averaged over the number of cells within the network.

1.2 .Cellular Communication Networks (CCNs)

One of the most popular and spreading applications of wireless communication networks is CCNs (also known as mobile telephone services). During the last two decades CCNs have shown a revolutionary growth and a rapid development. The essence of a CCN is the use of multiple low-power transmitters, on the order of 100 Watt or less. Because the range of such a transmitter is small, an area can be divided into cells, each one served by its own antenna. Each cell is allocated a band of frequencies (channels); and is served by a BS, consisting of transmitter, receiver, and control unit. Adjacent cells are assigned different frequencies to avoid interference or crosstalk.

However, cells sufficiently distant from each other can use the same frequency band. Figure (1.1) shows CCN components [Ahm 06, Fel 06, Yan 05].

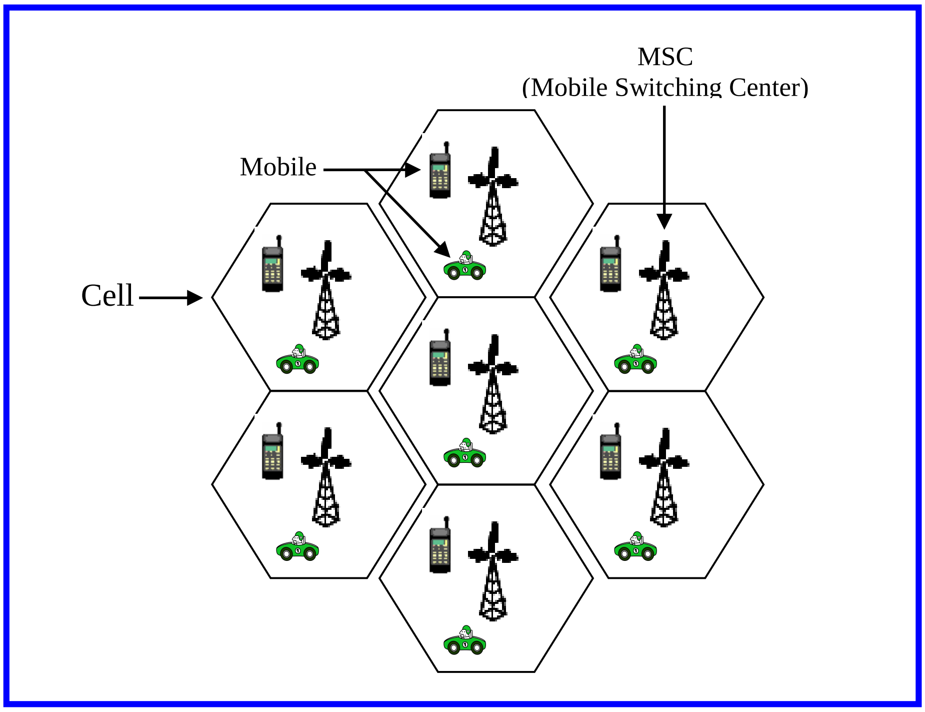


Figure (1.1). Cellular communication network (CCN).

1.2.1 Features of CCNs

There are a number of features that identify CCNs, such as [Fel 06, Lei 06, Lei 06]:

- The total coverage area is divided into cells, only a subset of channels is available in each cell.
- All available channels in the system are partitioned into sets; the sets are assigned to cells, two cells could be assigned the same set of channels if the two cells are separated by a sufficient geographical distance (minimum reuse distance) so that interference is small.
- One antenna per cell.
- Antennas are controlled by Mobile Switching Centers (MSC).
- Cells are modeled as hexagons.
- Cells interfere with each other.
- A mobile communicates with one (or sometimes two) antennas.
- To increase the capacity of the network, the cell area should be minimized to the minimum acceptable area and consequently the number of cells for the same network area increases.

1.2.2 Advantages and disadvantages of CCNs

CCNs have many advantages over other networks (e.g., Improved Mobile Telephone System (IMTS)), such as: they have more capacity due to spectral reuse, and lower transmission power due to smaller transceivers distances. They are more robust systems as a problem in any BS only affects the immediate cell. In addition, they have more predictable propagation environment due to shorter distances [Fel 07, Rob 00, Tan 03].

On the other hand, CCNs suffer from a number of disadvantages, examples of such disadvantages may include: need for more infrastructure, need for fixed network to connect BSs, some residual interference from co-channel cells, handover procedure required, and bandwidth limitations.

1.2.3 The Global System for Mobile communications (GSM)

In the early 1980s there were analogue technologies, e.g., Advanced Mobile Phone Services (AMPS) in North America, Total Access Communications System (TACS) in the UK, and Nordic Mobile Telephone (NMT) in Nordic countries. Since each country developed its own system, a number of problems have stimulated, such as [Nav 02, Tan 03]:

- The system worked only within the boundaries of each country.
- The mobile equipment manufacturers markets were limited by the operating system.

The solution was the Global System for Mobile communications (GSM), which was developed by the Conference of European Posts and Telecommunications (CEPT). GSM is an open digital cellular technology used for transmitting voice and data services. It differs from first generation wireless systems in that it uses digital technology and TDMA techniques, and more recently, GSM uses a CDMA technique. The details and specifications of GSM will be discussed in Chapter 3.

1.3. Channel Allocation

The use of wireless communication networks is experiencing a revolutionary growth throughout the world. The growth is fuelled by the expansion of CCNs, progress in data communications, and the spectacular development of the Internet. This rapid growth in the size of cellular networks, and the complexity of applications and offered services provide numerous challenges in the design and study of cellular systems.

As it has been mentioned earlier, channel allocation is a major component of network utilization and QoS provisioning in such systems. Because of scarcity of channels (Finite spectrum), the demand for better QoS (applications require more and more bandwidth), low capacity channels, and interference among users, optimal channel allocation problems play an ever-increasing role in the design and analysis of cellular networks.

Channel allocation could be defined as, how to determine who gets to use the channel when there is competition for it, so channels have to be used efficiently to utilize the network to its maximum potential. In cellular networks, desirable channel allocation is mainly achieved by the BS, through connection admission decisions, in the presence of new connection requests. Moreover, a key requirement is that the channels resources should not all be gained from the same central location in the cellular network, because a centralized resource would give a central point of failure if it fails.

These are challenging problems, thus, practical methods must be found for allocating the scarce channels that satisfy users adequately and efficiently.

1.3.1 Desirable characteristics of channel allocation techniques

There is a number of desirable characteristics that any satisfactory channel allocation techniques should have, such as [Gar 06, Jia 08, Ras 07, Zha 89]:

i. Minimum connection set-up time.

ii. Adaptable to changing load distribution.

iii. Fault tolerance.

iv. Scalability.

v. Low computation and communication overhead.

vi. Minimum handoffs.

vii. Maximum number of calls that can be accepted concurrently.

1.3.2 Channel allocation techniques

Many channel allocation techniques have been proposed during the last two decades for infrastructure wireless networks to avoid channel interference, efficiently utilize the limited frequencies (bandwidth) available, provide an adequate QoS for the mobile users, reduce the rate of dropped calls, etc [Cao 03, Pic 07].

These techniques can be classified into three different categories; these are [Abe 05, Kha 08, Leu 07, Mah 07]:

i. Static Channel Allocation (SCA)

ii. Dynamic Channel Allocation (DCA)

iii. Hybrid Channel Allocation (HCA)

Following is a brief description for each of the above techniques.

i. Static Channel Allocation (SCA)

Static channel allocation (SCA) systems, alternatively referred to as fixed channel allocation (FCA) systems, allocate specific channels to specific cells. In SCA techniques the allocated channels can not be changed. However, for efficient operation, SCA systems typically allocate channels in a manner that maximizes frequency reuse. Thus, in a SCA system, the distance between cells using the same channel is the minimum reuse distance for that system.

The problem with SCA systems occurs whenever the offered traffic to a cellular network is not uniform, considering a case in which two adjacent cells are allocated *f* channels each. There clearly can be situations in which one cell has a need for *f*+*k* channels while the adjacent cell only requires *f-m* channels (where both *k* and *m* are positive integers). In such a case, *k* users in the first cell would be blocked from making calls while *m* channels in the second cell would go unused. Clearly in this situation of non-uniform spatial offered traffic, the available channels are not being used efficiently. Despite this fact, SCA has been implemented on a widespread level to date [Abe 05, Wan 07].

ii. Dynamic Channel Allocation (DCA)

Dynamic channel allocation (DCA) attempts to alleviate the problem mentioned for SCA systems when offered traffic is non-uniform. In DCA systems, no set relationship exists between channels and cells. Instead, channels are part of a pool of resources. Whenever a channel is needed by a cell, the channel is allocated under the constraint that frequency reuse requirements can not be violated.

There are two problems that typically occur with DCA based systems, these are [Lee 07, Ver 07, Zha 89]:

1. DCA methods typically have a degree of randomness associated with them, and this leads to the fact that frequency

2. reuse is often not maximized, unlike the case for SCA systems in which cells using the same channel are separated by the minimum reuse distance.

3. DCA methods often involve complex algorithms for deciding which available channel is most efficient. These algorithms can be very computationally intensive and may require large computing resources in order to be real-time. However, this problem has been overcome with high increase of the speed of processing systems.

There are mainly two types of DCA schemes:

1. Centralized DCA (CDCA). The CDCA scheme involves a single controller selecting a channel for each cell. CDCA schemes can theoretically provide the best performance. However, the enormous amount of computation and communication among BSs leads to excessive system latencies and renders CDCA schemes impractical.

2. Distributed DCA (DDCA). The DDCA scheme involves a number of controllers scattered across the network. In DDCA, a

3. BS communicates with each other without any central control to find the channel that does not interfere with the neighboring cells.

The main features of the SCA and the DCA are compared in Table (1.1) [Cao 03, Kog 07].

iii. Hybrid Channel Allocation (HCA)

In hybrid channel allocation (HCA) scheme, each cell has a static channel set permanently allocated to it and a reserved pool of channels that can be borrowed. This scheme is a trade off between SCA and DCA and so are their merits and demerits [Cao 03, Kog 07].

Table (1.1)
A comparison between the static and the dynamic channel allocation techniques.

DCA	SCA
Performs better under light or moderate traffic	Performs better under heavy traffic
Not always maximum channel reusability	Maximum channel reusability
Insensitive to time and time spatial changes	Sensitive to time and spatial changes
Suitable in micro cellular environment	Suitable for large cell environment
High flexibility	Low flexibility
High computational effort	Low computational effort

Moderate to high implementation complexity	Low implementation complexity
Centralized, distributed control depending on scheme	Centralized control

1.4.Statement of the Problem

Due to the limited available recourse in CCNs, efficient channel allocation strategy with simultaneous fulfilment of better bandwidth utilization, minimum interferences, minimum communication overheads, highest possible QoS, and satisfying all other desirable characteristics of channel allocation techniques in CCNs, are still critical and important practical issues.

A number of techniques have been developed throughout the years for efficient channel allocation in CCNs, but each of them has its own drawbacks and limitations (e.g., slow convergence, infinite loop, high communication overheads, etc.). Thus, these techniques are still short from satisfying the needs of the mobile CCNs users and service providers.

In this thesis, we develop and evaluate the performance of an efficient and a reliable DCA scheme that can be used for channel allocation in CCNs, and convene all requirements and constraints imposed by the user, service providers, and technology of CCNs. The scheme is based on the AWC algorithm, which is, in turn, based on a formalism that is widely used for various application problems in Distributed Artificial Intelligence (DAI), called Distributed Constraint Satisfaction Problem (DCSP). In order to minimize the data communication overheads of the AWC algorithm, a token-based mechanism is used.

1.5. Objectives of this Work

The main objectives of this work are:

1. Develop and evaluate the performance of an efficient and a reliable solution for DCA in CCNs.
2. Evaluate the performance of the ABT and the AWC algorithms for solving DAI problems that are formalized as DCSP, such the well-known *N*-Queen puzzle.
3. Demonstrate the effectiveness of the AWC algorithm for solving DCA problem in CCNs to achieve optimal bandwidth utilization under different network operation environments.

1.6 Thesis Organization

The rest of this thesis is organized as follows. Chapter 2 presents a literature review that summarizes the most recent and related work. It is divided into two sections; the first section reviews work that is related to the DAI techniques, and the second section presents a review on the work that is related to a number of efficient and reliable DCA schemes.

Chapter 3 provides an introduction to the CCNs and GSM systems. In Chapter 4, a detailed description of the new DCA scheme is given. It also provides an introduction to DAI, CSP, and DCSP. In addition, in Chapter 4, we present the concept, methodology, and implementation of the ABT and the AWC algorithms. In order to validate the accuracy of the implementation and evaluate the performance of these two algorithms, they are used to solve a simple standard AI benchmark problem, namely, the *N*-Queens problem.

In Chapter 5, a number of experiments or simulations are described and carried-out to demonstrate and evaluate the performance of the new DCA scheme in solving realistic resource allocation problems in CCNs exemplified by a GSM system. The results obtained are presented in tables and graphs, and the performance of the DCA scheme is compared. Finally, in Chapter 6, conclusions are drawn and recommendations for future works are pointed-out.

Chapter 2
Literature Reviews

2.1. Introduction

A number of mechanisms have been developed throughout the years to solve the channel allocation problem in cellular networks, to make the system highly adaptive to traffic changes, to utilize the available spectrum efficiently, and to allocate channels optimally to the cells within the network. These mechanisms mainly include:

i. Static channel allocation (SCA), also called fixed channel allocation (FCA)

ii. Dynamic channel allocation (DCA)

iii. Hybrid channel allocation (HCA)

iv. Distributed Artificial Intelligence (DAI) techniques for channel allocation.

In this chapter, some of the most recent work that is related to both the DCA schemes and the DAI techniques are reviewed. Section 2.2 presents a review on the work that is related to the DAI techniques, while the work that is related to the DCA schemes is presented in Section 2.3.

2.2. DAI Techniques

I. Emary [Ema 06] concerned with a practical application of DAI for managing the high data rate bus structured local area computer network that uses deterministic multiple access protocol. In the selected network that is managed using DAI, the dynamic sharing of the available bandwidth among stations is achieved by forming “train to which each station may append a packet after issuing a reservation. The managing approach depended on using intelligent autonomous agents, which are responsible for various tasks among it: election of the end stations, the recovery from failures, and the insertion of new stations in the network. All these tasks are based on the use of special tokens.

However this study is just a theoretical approach that was not operated practically on a network or tested using any simulator, moreover it was studied for a wired network with a centralized controller.

G. Abeysundara [Abe 05] proposed a dynamic cellular channel allocation using intelligent agents. Under his implementation, agents only interact with the environment and the network cells. An aspect of self-organization is the reliance on multiple interactions, and the ability of agents to make use of the results of their own actions and the actions of others. The latter is not very apparent in his system; agents make very little use of the actions of others.

W. Zhang et. al [Zha 05] formalized the distributed scheduling problems in distributed sensor networks, in which computational and communication resources are scarce, as DCSPs and distributed constraint optimization problems (DCOPs) and modeled them as distributed graph coloring.

They found that to cope with limited resources and restricted real-time requirement, it is imperative to use distributed algorithms that have low overhead on resource consumption and high-quality anytime performance. In order to meet these requirements, they studied two existing DCSP algorithms, distributed stochastic search algorithm (DSA) and distributed breakout algorithm (DBA), for solving DCOPs and the distributed scheduling problems.

They experimentally showed that DSA has a phase-transition or threshold behavior, in that its solution quality degenerates abruptly and dramatically when the degree of parallel executions of distributed agents increases beyond some critical value. They also considered the completeness and complexity of DBA for distributed graph coloring, where they demonstrated that DBA is complete on coloring acyclic graphs. To improve DBA's performance on coloring cyclic graphs, they proposed two stochastic variations. Finally, they directly compared DSA and DBA for solving distributed graph coloring and distributed scheduling problems in sensor networks. Their results showed that DSA is superior to DBA when controlled properly, having better or competitive solution quality and significantly lower communication cost than DBA. Therefore, DSA is the algorithm of choice for distributed scheduling problems and other distributed problems of similar properties.

R. Bejar et al. [Bej 04] reported an experimental study of the average-case computational complexity of two early algorithms, ABT and AWC search on an application in distributed sensor networks. They also showed that random effects, both intentional such as random value selection and unintentional such as random delays, have a significant effect on the performance of the algorithms. Finally, they pointed-out that there are big performance differences between solvable and unsolvable instances.

C. Bessiere et al. [Bes 04] proposed a new ABT algorithm for DCSPs named ABTnot. They showed that his procedure was the first one that did not add links between agents not sharing constraints. This property can be important to avoid messages sent to agents which may not need to be informed, which of course will have a significant effect on the system efficiency.

H. Jung and M. Tambe [Jun 04] investigated the impact of inter-agent exchange of additional information within a collaborative setting. They provided a new run-time model for DCSP performance measurement that takes into account the overhead of extra communication in various computing and networking environments. They showed that exploiting additional information-exchange can improve performance in a significant range of problem settings and also provided categorization of problem settings with big speedups by the DCSP strategies to guide strategy selection.

M. Silaghi and B. Faltings [Sil 04] showed that asynchronous search algorithms for DCSP so far have taken little advantage of the techniques that have led to highly efficient centralized algorithms for CSP. They had shown how two well-known techniques, namely, value aggregation and arc consistency, can be used to significantly improve the performance of distributed asynchronous search algorithms.

R. Wallace and E. Freuder [Wal 04] showed that there are potentially important tradeoffs between maintaining privacy and enhancing search efficiency in multi agent systems. In their work they showed how quantitative assessments of privacy loss can be made within the framework of DCSP. They also showed how agents can make inferences about other agents' problems or sub problems from communications that carry no explicit private information.

This can be done using constraint-based reasoning in a framework consisting of an ordinary CSP, which is only partly known, and a system of shadow CSPs that represent various forms of possibilistic knowledge. This kind of reasoning in combination with arc consistency processing can speed up search under conditions of limited communication, at the same time potentially undermining privacy.

P. Modi et al. [Mod 01] proposed a formalization of distributed resource allocation that is expressive enough to represent both dynamic and distributed aspects of the problem. They showed that this approach to dynamic and distributed resource allocation is powerful and unique, and can be applied to real-problems such as the distributed sensor network domain (DSND). The central contribution of their work was a generalized mapping from distributed resource allocation to dynamic DCSP.

S. Willmott [Wil 01] argued that in combination with careful use of abstraction, DAI coordination, organization and organizational design, techniques can be used to build efficient solutions to certain classes of distributed control problems, concretely the challenging problem of on-line, state-based QoS routing in a fully distributed simulation.

He demonstrated that this new approach has significant advantages over the static hierarchical routing approaches that are often proposed for this type of QoS routing. He also identified potential difficulties with the dynamic behaviour of his approach.

K. Al-Agha [Agh 00] proposed a multi-agent solution for intelligent BSs in wireless networks to verify its feasibility as a main target. He found agents are able to combine knowledge and experience with neighbouring agents to make the best decisions, also he demonstrated that, the intelligent agent approach to introduce the self-adaptive resource allocation feature in mobile networks remains an attractive and formal way of integrating intelligence in BSs.

M. Yokoo and K. Hirayama [Yok 00] provided an overview of the existing research on DCSPs. They explained the formalization of several multi-agent systems (MAS) application problems using DCSPs. They described a series of algorithms for solving DCSPs, namely the ABT, the AWC, the distributed breakout, and distributed consistency algorithms.

They demonstrated that one limitation of the ABT algorithm was that the agent or variable ordering is statically determined. If the value selection of a higher priority agent is bad, the lower priority agents need to perform an exhaustive search to revise the bad decision. For the AWC search algorithm, they introduced the min-conflict heuristic to reduce the risk of making bad decisions and the agent ordering is dynamically changed so that a bad decision can be revised without performing an exhaustive search. They compared the search algorithms for solving DCSPs with a single local variable. They found that the AWC search outperforms the ABT, also they showed that the distributed breakout outperforms the AWC search when problem instances are critically difficult, but AWC outperforms the distributed breakout when problem instances are sparse.

2.3 .Dynamic Channel Allocation (DCA) Techniques

R. Verdone et. al presented the impact of user mobility and the handover process on the performance of interference adaptive (IA) dynamic channel allocation (DCA) schemes. The results have been achieved by means of a simulation tool that took propagation, mobility, handover, power control, directed retry (DR), interference, traffic, and channel allocation into account.

Moreover, some simplified analytical descriptions were given to discuss the behaviour of the blocking probability when varying the speed of users. They compared the performance of well-known totally distributed IA-DCA schemes to that of a partially distributed algorithm. The comparison to fixed channel allocation was also considered, two different mobility models were considered too; the numerical results showed that the satisfaction probability changed when taking user mobility into account, whereas the blocking probability was scarcely affected by the mobile speed; the latter statement was compared to previous results from the literature.

F.P. Garcia et al. [Gar 06] focused on the number of mobile devices expected to be within the area of coverage of each BS at a given moment in the future. They proposed an indexing mechanism for indexing and retrieving aggregate information on mobile devices, based on the indexation structures of the TPR-Tree (Time-Parameterized R-Tree) and the aR-Tree (aggregate R-Tree) by which aggregate time slice queries (ATQ) can be performed in order to estimate the number of mobile devices expected to be within each BS area at a given moment. Once an ATQ had been performed, DCA algorithms may be employed to reduce the total number of communication channels available to a cellular network and thereby reducing implantation and maintenance costs.

Also they proposed the use of historical aggregate time slice queries (HATQ) to provide the number of mobile devices expected to be within each BS area between two given moments in past time. They concluded, when historical data are used in DCA algorithms together with results from ATQ (estimates for the future) channel allocation could be anticipated with more precision.

S. Tokekar and N. Purohit [Tok 06] presented an FCA algorithm in scenarios where one or more low traffic cells are located with heavy traffic surrounding cells, and the size of the cells in the system is not equal. They concluded that if the borrowed channels are locked in the worst affected first tier co-channel cells, then to meet the traffic requirement in these cells the channels can be borrowed from their adjacent sectors.

They claimed that grouping and sub grouping of channels plays a very important role in the overall performance of the system, and depending upon the structure of a particular network, an optimum way can be found to allot the channels to various BTs or to reshuffle them. Also adjacent channel interference is another important parameter which was needed to be considered in grouping the channels with the proposed scheme.

J. Yang et al. [Yan 05] proposed an efficient fault-tolerant channel allocation algorithm which achieves high channel utilization. In the proposed algorithm, a cell may borrow a channel even it receives some partial channel information from some of its neighbours. Moreover, a cell can lend a channel to multiple borrowers (at most three) as long as any two of them are not neighbours.

R. Babbar et al. [Bab 04] presented an agent based scheme for efficient management of radio resources in hybrid wireless networks. Performance of the proposed scheme is measured in terms of successful handover rate between different wireless network architectures (e.g., WAN, Cellular), and also by the allocated bandwidth to admitted calls. Simulation results showed that the proposed agent-based approach provides a 10% increase in the average allocated bandwidth obtained with conventional resource management schemes.

Z. Kostic and N. Sollenberger [Kos 02] evaluated the performance of dynamic frequency hopping (DFH) when applied to cellular systems with a limited total bandwidth. They also illustrated a practical implementation for DFH deployment using network-assisted resource allocation (NARA). The performance evaluation was accomplished by system-level simulations of a system with 12 carriers and 1/1 frequency reuse, based on the EDGE-Compact specification.

They assumed a voice-only circuit-switched operation, and they modeled fading channel, multicell interference, voice activity, and antenna sectorization. They presented the performance of dynamic frequency hopping compared to random frequency hopping (RFH) and fixed channel assignment by showing distributions of word error rates.

They studied sensitivity to occupancy, Rayleigh fading assumptions, number of carriers, voice activity, and measurement errors. They also compared the uplink and downlink performance. The results they obtained indicated that DFH can significantly improve the performance compared to random frequency hopping. For example, at a 2% frame error rate with 90% coverage, the capacity improvement of DFH is almost 100% when compared with fixed channel assignment, and about 50% when compared to RFH. Finally, they concluded that the amount of improvement for the uplink direction is smaller than the improvement for the downlink direction, especially for higher occupancies.

Z. Kostic et. al [Kos 01] examined techniques for increasing spectral efficiency of cellular systems by using slow frequency hopping (FH) with dynamic frequency-hop (DFH) pattern adaptation. They first presented analytical results illustrating the improvements in frequency outage probabilities obtained by DFH in comparison with random frequency hopping (RFH). Next, they showed simulation results comparing the performance of various DFH and RFH techniques. System performance was expressed by cumulative distribution functions of codeword error rates. Systems that were studied incorporate channel coding, interleaving, antenna diversity, and power control. Their analysis and simulations considered the effects of path loss, shadowing, Rayleigh fading, co-channel interference, coherence bandwidth, voice activity, and occupancy. The results indicated that systems using DFH can support substantially more users than systems using RFH.

M. Salmenkaita et al. [Sal 01] presented a practical dynamic channel assignment scheme for cellular GSM networks that is called a dynamic frequency and channel assignment (DFCA) scheme. They showed that the behaviour of DFCA was satisfactory in very high load situations where the gain remains stable or is even increasing as happened with 50 km/hr mobile speed. In their model, a BS is capable of base band frequency hopping, and can only utilize part of the DFCA frequency band, therefore limiting the freedom of radio channel selection. Also they proved that in a typical case the gain of DFCA was reduced in a linear manner as the share of base band hopping BSs increases. Also DFCA can also be used to provide radio channel differentiation based on connection type, and this can be utilized to maximize the network performance when the frequency reuse does not have to be dimensioned from the worst case point of view.

S. Boumerdassi [Bou 00] proposed an adaptation for DCA of variable channel reservation scheme proposed first for FCA. His new mechanism which adjusted the resource reservation in the pool according to the fluctuation on the traffic. DCA variable reservation (DVR) enables significant improvement of the grade of service (GoS). The required guard bandwidth can be calculated off-line and loaded into lookup tables to facilitate the dynamic allocation.

X. Wu et al. [Wu 00] expanded a channel borrowing (CB) scheme and proposed a new channel allocation scheme called mobile-assisted connection-admission (MACA) scheme, to achieve load balancing in a cellular network. Using MACA scheme, the authors proposed some special channels that are used to connect mobile units from different cells; thus, a mobile unit, which is unable to connect to its own BS because it is in a heavily-loaded “hot” cell, may be able to get connected to its neighbouring cold cell’s base station through a two hop link. They found that MACA can greatly improve the performance of a cellular network over FCA scheme.

S. Yubin et al. [Yub 00] discussed the problem of DCA in cellular communication networks. They developed a simple but useful method to calculate the lower limit of call blocking probability of DCA, this method could be used to compare the performance of FCA with any kind of DCA schemes easily and clearly. They found that the lower limit of blocking probability of DCA is related to the cluster size *N*, the lower limit of blocking probability of DCA will decrease if *N* increases. For example, they proved that in GSM system the application of DCA strategy would greatly improve the overall system capacity, while in CDMA systems in which *N* is less than 3 the improvement by DCA is not so obvious.

Y. Zhang [Zha 99] proposed and evaluated a new channel allocation algorithm and compared it with the load balancing strategy with selective borrowing (LBSB) algorithm, in mobile cellular systems using simulation. He found that the new algorithm improves the call blocking probability and the implementation complexity over the LBSB algorithm. He reasoned his result by the following:

i. Each cell can communicate with the MSC autonomously in the algorithm so that the message exchanges between the MSC and the cells processed in parallel.

ii. The new algorithm used a two-threshold scheme which prevents a cell from ping-pong state changes (switched back and forth between cold and hot states, resulting useless channel borrowing and lending).

iii. The algorithm took into account the co-channel cells of a lender cell in making borrowing decisions so that a potential channel borrowing loop was avoided.

The algorithm adjusted adaptively the values of the light and the heavy thresholds according to the system state so that better borrowing decisions can be made.

W. Wan and W. Wong [Wan 98] investigated a multi-slot allocation algorithm for multimedia data on a hybrid TDMA/FDMA digital cellular system. They employed a mobile unit with the capability of transmission through multiple frequency bands. Their idea was, instead of searching for all combinations of frequencies to obtain enough idle time slots as in the locally optimal algorithm, a heuristic algorithm based on binary clustering of idle slots was proposed. They showed that the inference property of the algorithm speeds up the searching procedure. The simulations were done to investigate the performance of both the locally optimal algorithm and the proposed algorithm. They found that the proposed algorithm was much more time efficient with minimal trade-off.

L. Guerrero and D. Rodriguez [Gue 97] evaluated Compact Pattern with Maximized Channel Borrowing (CPMCB) with three schemes for channel allocation in an interference limited environment under real non uniform traffic conditions in Mexico City, taking into account the pan European GSM standard. The three schemes are FCA, Borrowing with Channel Ordering (BCO), and Borrowing with Directional Channel Locking (BDCL). The results were obtained on an asymmetrical tele-traffic load situation with the GSM standard.

The results showed that the average blocking probability as the based load in the system was increased. When a non uniform traffic pattern is used to evaluate the strategies and when the based load is increased from 1 to 10 times (1000%) for the GSM standard, CPMCB outperforms BDCL and BCO for a blocking rate of about 1.6% to 2.5%, the advantage in traffic carrying capacity is rather low (1.33%) with respect to BDCL but it is about 5.5% in comparison with BCO.

When the blocking rate of the central cluster is taken into account, CPMCB shows an increase of 37.5% and 16% in comparison with BDCL and BCO, respectively, at a blocking rate of 2%. CPMCB is outperformed only by BCO and BDCL when the load increase is more than 8.5 times the based load. With CPMCB, the grade of service of 0.02 is met at higher traffic in comparison with any other strategy at a constant offered load.

To get an insight into the amount of switching or channel reallocations per call occurring in each of the schemes, for this case, the number of channel reallocations for CPMCB is always inferior to those carried by BDCL and BCO, decreasing the signalling load of the system.

F. Priscoli et al. [Pri 97] considered the impacts of the introduction of the DCA strategies in the GSM network taking a realistic scenario as a reference. They showed that the DCA strategies require a small amount of information, and are compatible with the GSM procedures. They particularly studied the geometric DCA (GDCA) and the cost function DCA (CFDCA), which are requiring a different amount of information. They demonstrated how the information exchanges among base transceiver stations (BTSs) can be implemented, how conflicting carrier acquisitions can be avoided, and how the transmitting chains can be implemented so that they can be tuned in real time, on the carrier selected by the DCA algorithm.

All of the provisions suggested for solving the above-mentioned problems were feasible, and were compatible with the GSM method of operation. The performance achievable with the considered DCA strategies was assessed by simulations which have taken into account all of the main GSM transmission and network aspects in a real context (realistic cell layout, propagation data, traffic model, etc.). On the basis of the performance results, it was concluded that the DCA implementation is actually worthwhile since it entailed an increase in the number of mobile users supportable by the cellular network on the order of 40–50% with respect to the FCA.

Even more remarkable advantages for the DCA strategies are expected when considering non constant average traffic per cell. As for the comparison between the GDCA and the CFDCA, it was shown that the CFDCA has a slight advantage in terms of performance over the GDCA, but the CFDCA required a higher amount of real-time information, thus entailing some additional signalling overhead among BSs. On the other hand, the GDCA requires the knowledge of some semi permanent information which is not necessary for the CFDCA.

O. Yu and V. Leung [Yu 97] proposed a novel dynamic guard channel scheme which adapts a number of guard channels in each cell, according to the current estimate of the handoff call arrival rate derived from the current number of ongoing calls in neighbouring cells and the mobility pattern, so as to keep the handoff call blocking probability close to the targeted objective, while constraining the new call blocking probability to be below a given level.

The proposed scheme is applicable to channel allocation over cellular mobile networks, under stationary traffic conditions, and it offers better resource utilization than that of the fixed guard scheme by providing lower and comparable long-term handoff call blocking probabilities under respective medium and heavy bandwidth loadings. Under non stationary traffic conditions, the proposed dynamic guard scheme enables the handoff call blocking probability to stay close to its targeted objective, without significantly increasing the new call blocking probability relative to the fixed guard scheme.

C. Carciofo et al. [Car 96] studied dynamic allocation schemes in a realistic environment, they focused on the GSM cellular network in the metropolitan area of Rome, and they investigated the feasibility of traffic and interference adaptive resource allocation schemes in a realistic GSM and DCS 1800 cellular environment. They had verified that both these schemes enhanced the performance of GSM and DCS 1800 systems under non-uniform realistic traffic distribution conditions. When power control and direct retry are applied, traffic-adaptive schemes offered an improvement in terms of system capacity in the range 40-60% depending on the desired constraints on blocking and dropping probabilities.

A preliminary evaluation of the signalling introduced by traffic adaptive schemes had been also carried out and they concluded that the signalling load requested by these algorithms does not seem to introduce a significant overhead with respect to FCA. On the contrary the segregation interference-adaptive algorithm does not require any new signalling between BSs and an improvement of system capacity in the range 30-60% can be achieved.

Chapter 3
Cellular Communication Networks and GSM Systems

3.1.Introduction

The main objective of this work is to develop and evaluate the performance of a Dynamic Channel Allocation (DCA) scheme that is based on a Distributed Artificial Intelligent (DAI) algorithm, namely, the Asynchronous Week-Commitment (AWC) algorithm for resource allocation in Cellular Communication Networks (CCNs) (e.g., the Global System for Mobile communication (GSM)). Before we proceed with the description, development, and evaluation of the DCA scheme, we devote this chapter to provide a theoretical and technological background for both CCNs and GSM systems.

Section 3.2 presents an introduction to CCNs, the possible cell shapes, the main features and advantages of the hexagonal cell shape that is widely used in mobile phone systems, the frequency reuse and why it is necessary, approaches for increasing CCNs capacity, and CCNs technologies in use. An insight into the design and operation of the GSM is given in Section 3.3. It gives the history of GSM, the services provided by GSM, the GSM system architecture, the GSM frequency bands, and the radio transmission and network aspects.

3.2. Cellular Communication Networks (CCNs)

A CCN is a radio network made up of a number of radio cells each served by a fixed transmitter, known as a cell site or base station (BS). Cellular networks are considered as one of the most popular and spreading applications for wireless networks. During the last two decades cellular networks have shown a huge revolutionary growth and a rapid development. The essence of a CCN is the use of multiple low-power transmitters. Because the range of such a transmitter is small, an area can be divided into cells. Each cell is allocated a band of frequencies; and is served by a BS, consisting of transmitter, receiver, control unit, and antenna. Adjacent cells are assigned different frequencies to avoid interference or crosstalk. However, cells sufficiently distant from each other can use the same frequency band.

CCNs offer a number of advantages over alternative solutions, such as [Gar 06, Lee 07, Leu 07, Sta 05]:

- increased capacity
- reduced power usage
- better coverage

- increased local availability
- low cost and ease of maintainability

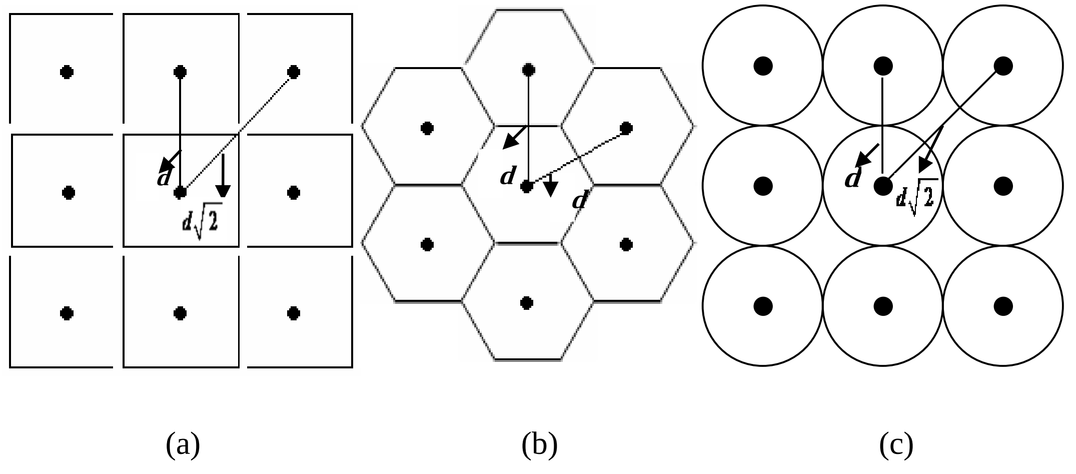
3.2.1 Cell shape

Determining the shape of the cell in a cellular system is an important matter, because the cell shape will determine the area to be covered, the number of adjacent neighbours for the cell, the distance of each neighbour from the cell, and if there are gaps or overlaps areas.

If the shape of the cell that covers an area is circular then there will be gaps that are not covered by any cell, and also there will be some area that may be covered by more than one cell which will lead to an overlapped area between two cells or more, this has unwanted consequences such as frequency interference and more than one BS that controls the overlapped area.

As mentioned before the shape of the cell determines the distance of each neighbour from the cell, in this case there could be a cell that has neighbours at different distances from that cell, this could lead to a case that if there is a mobile user within a cell moving toward the cell's boundaries, it is best if all of the adjacent antennas are at equal distance from the mobile user cell, this simplifies the task of determining when to switch the user to an adjacent antenna and which antenna to choose.

In Figure (3.1), we compare between three types of cell shapes, namely, a square cell, a hexagonal cell, and a circular cell.



(a) (b) (c)

Figure (3.1). Cell shapes [Sta 05, Tan 03].

In Figure (3.1a) where the cell shape is square, if the width of the square is *d* then the cell has four neighbours at distance *d* and four neighbours at distance $d\sqrt{2}$. Figure (3.1b) shows a hexagonal uniform cell shape, in which each opposite side in the hexagonal cell is separated by a distance *d*, thus each cell has six neighbours at distance *d*. A circular cell shape of a diameter *d* is shown in Figure (3.1c). In this figure it can be seen that each cell has 8 neighbours, four of them at distance *d* and four at distance $d\sqrt{2}$, In other words it is similar to the square cell shape, but also there are areas that are not covered by a transmission from any cell.

3.2.2 Characteristics of a hexagonal cell

A hexagonal cell shape has many advantages that motivated its usage in cellular networks, hexagonal cells cover an area without gaps or overlaps, that's because hexagonal cell is closest to a circle without having gaps or overlaps, about the hexagon radius *r*, it is defined to be the radius of the circle that circumscribes it, which equals the distance from the hexagon center to each vertex, which also equals the length of a side of a hexagon.

As shown in Figure (3.2), for a cell of radius *r*, the distance between two adjacent cells' centers is *d*, which is given by:

$$d = \sqrt{3} \times r \qquad (3.1)$$

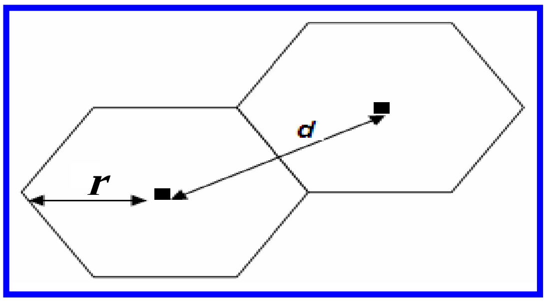


Figure (3.2). Hexagonal cell.

As a result the hexagon center (where the antenna should be placed) is at equal distance from each adjacent hexagon center, so a mobile user at the boundaries of some cell area will have a little calculations to decide to which adjacent cell he should move to.

3.2.3 Frequency reuse

In a cellular system, each cell has a base transceiver. The transmission power is carefully controlled to allow communication within the cell using a given frequency band while limiting the power at the frequency that escapes the cell into adjacent cells. Nevertheless, it is not practical to attempt to use the same frequency band in two adjacent cells. Instead, the objective is to use the same frequency band in multiple cells at some distance from one another. This allows the same frequency band to be used for multiple simultaneous conversations in different cells. Multiple frequency bands are assigned, generally, 10 to 50 frequencies are assigned depending on the traffic expected [Mah 07, Spy 07, Sta 05].

There are many reasons for the need of frequency reuse, such as:

1. An increase in demand and the poor quality of existing service led to research into ways to improve the QoS and support more users in systems.

2. Frequency spectrum available for mobile cellular use was limited; therefore efficient use of these frequencies was necessary, so the radio channels must be reused to carry more than one conversation at a time.

Frequency reuse could be done by using patterns (clusters), the pattern consists of a number of cells, each cell in the pattern uses a unique set of frequency bands.

The number of frequencies that can be allotted for each cell in the pattern can be calculated by:

$$f = \frac{K}{N} \tag{3.2}$$

Where:

f the number of frequencies allotted for each cell.

K the total number of frequencies available for the system.

N the number of cells in a repetitious pattern.

After that this pattern is repetitious, which means the other cells in the system will be organized into other patterns of N cells that can provide sufficient isolation between two uses of the same frequency, and the repetition of the pattern depends on the number of cells in the system, the pattern will be repeated to contain all the cells in the system.

In characterizing frequency reuse, the following parameters are commonly used:

D is the minimum distance between centers of cells that use the same frequency band (called co-channels).

r is the radius of a cell.

d is the distance between centers of adjacent cells ($d = \sqrt{3} \times r$).

N is the number of cells in a repetitious pattern (each cell in the pattern uses a unique set of frequency bands), termed the reuse factor.

A key design issue in cellular networks is to determine the minimum separation between two cells using the same frequency band, so that the two cells do not interfere with each other [Sta 05]. There are different possible patterns of frequency reuse. Figure (3.3) shows some examples. It can be easily proved that the smallest pattern that can provide sufficient isolation between two uses of the same frequency is 7, i.e., N=7. In addition, the following equations hold the relationship between D and r:

$N = 4$ $N = 7$

$$\frac{D}{r} = \sqrt{3N} \tag{3.3}$$

Substituting Eqn. (3.1) into Eqn. (3.3) yields:

$$\frac{D}{d} = \sqrt{N} \tag{3.4}$$

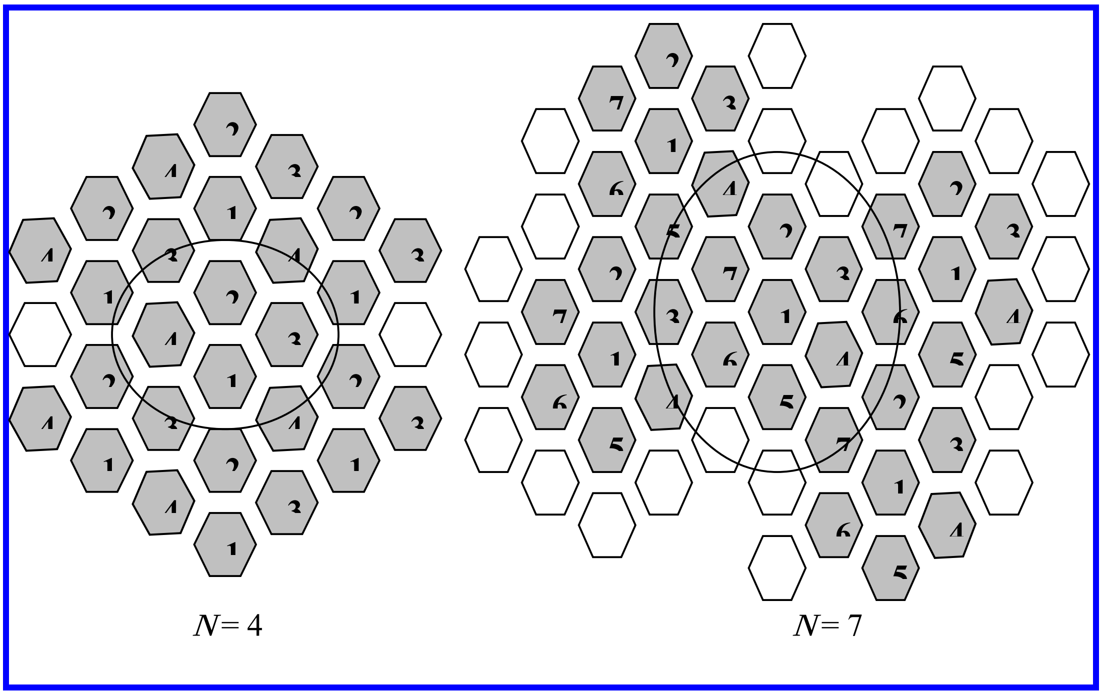


Figure (3.3). Patterns of frequency reuse [Sta 05, Tan 03].

3.2.4 Increasing network capacity

As time passes, more customers use the system, therefore, traffic may build up until there are not enough frequencies assigned to a cell to handle its calls. A number of approaches have been used to cope with this situation [Jia 08, Leu 07, Tan 02, Ver 07], such as:

- *Adding new channels*: Typically, when a system is set up in a region, not all of the channels are used, growth and expansion can be managed in an orderly fashion by adding new channels if available.

- *Frequency borrowing*: In the simplest case, frequencies are taken from adjacent cells by congested cells. The frequencies can also be assigned to cells dynamically.

- *Cell splitting*: In practice, the distribution of traffic and topographic features is not uniform, and this presents opportunities for capacity increase. Cells in areas of high usage can be split into smaller cells. To use a smaller cell, the power level used must be reduced to keep the original signal within the cell. Also, as the mobile units move, they pass from cell to cell, which requires transferring the call from one BS to another. This process is known as handoff.

- *Cell sectoring*: With cell sectoring, a cell is divided into a number of wedge-shaped sectors, each with its own set of channels, typically 3 to 6 sectors per cell. Each sector is assigned a separate subset of the cell's channels, and directional antennas at the BS are used to focus on each sector.

- *Microcells*: As cells become smaller, antennas move from the tops of tall buildings or hills, to the tops of small buildings or the sides of large buildings, and finally to lamp posts, where they form microcells. Each decrease in cell size is accomplished by a reduction in the radiated power levels from the BS antenna and the mobile units. Microcells are useful in city streets in congested areas, along highways, and inside large buildings.

Table (3.1) summarizes typical parameters for traditional cells, called macrocells and microcells with current technology. The average delay spread refers to multipath delay spread (i.e., the same signal follows different paths and there is a time delay between the earliest and latest arrival of the signal at the receiver). As indicated, the use of smaller cells enables the use of lower power and provides superior propagation conditions.

Table (3.1)
Typical parameters for macrocells and microcells [Mah 07, Sta 05, Tan 03, Web 6]

Parameter	Macrocell	Microcell
Cell radius	1 to 20 Km	0.1 to 1 Km
Transmission power	1 to 10 Watt	0.1 to 1 Watt
Average delay spread	0.1 to 10 □s	10 to 100 ns
Maximum bit rate	0.3 Mbps	1 Mbps

3.2.5 Cellular networks technology

The most common example of a cellular network is a mobile phone (cell phone) network. A mobile phone is a portable telephone which receives or makes calls through a BS. Radio waves are used to transfer signals to and from the cell phone.

Large geographic areas (representing the coverage range of a service provider) are split up into smaller cells to deal with line-of-sight signal loss and the large number of active phones in an area. Each cell overlaps other cell sites. All of the cell sites are connected to cellular telephone exchange switches, which in turn connect to the public telephone network or another switch of the cellular company.

As the phone user moves from one cell area to another, the switch automatically commands the handset and a cell site with a stronger signal (reported by the handset) to go to a new radio channel (frequency). When the handset responds through the new cell site, the exchange switches the connection to the new cell site.

With code division multiple access (CDMA), multiple CDMA handsets share a specific radio channel; the signals are separated by using a pseudo-noise code (PN code) specific to each phone [Kim 08, Spy 07]. As the user moves from one cell to another, the handset sets up radio links with multiple cell sites (or sectors of the same site) simultaneously. This is known as "soft handoff". It unlike “hard handoff”, which is used in traditional cellular technology, in which there is no one defined point where the phone switches to the new cell [Mar 07].

As it has been mentioned above, modern mobile phones use cells because radio frequencies are a limited shared resource. Cell-sites and handsets change frequency under computer control and use low power transmitters so that a limited number of radio frequencies can be reused by many callers with less interference. CDMA handsets, in particular, must have strict power controls to avoid interference with each other. An incidental benefit is that the batteries in the handsets need less power.

There is a number of different digital cellular technologies, including: GSM, General Packet Radio Service (GPRS), Code Division Multiple Access (CDMA), Evolution-Data Optimized (EV-DO), Enhanced Data Rates for GSM Evolution (EDGE), third-generation GSM (3GSM), Digital Enhanced Cordless Telecommunications (DECT), Digital AMPS (D-AMPS), and Integrated Digital Enhanced Network (iDEN) [Leu 07, Sar 00, Yan 05].

However, in this work we are mainly concerned with resource allocation in cellular networks. In particular, our experimental platform is the GSM system.

3.3 Minimum Interference Constraints in Cellular Networks

Radio signal interference depends on various parameters, such as cell shape, size, layout, defined protection ratio, applied modulation, etc. In general there are two types of interference in mobile cellular networks, these are:

i. *Co-channel interference:* occurs when two sufficiently close cells use the same channel simultaneously. To reduce this type

ii. of interference, channels should be reused simultaneously in cells which are sufficiently far apart so that an acceptable level of carrier-to-interferer ratio is maintained.

iii. *Adjacent-channel interference:* appears when a signal is deteriorated by interference caused by signal(s) in other radio channel(s) used in the same cell or in one or more other cells. The term "adjacent" originates from interference caused by usage of two channels adjacent in radio-spectrum in FDMA.

For classification purposes and depending on the type of interference which may occur between two cells, we distinguish two types of radio neighbours as shown in Figure (3.4), these are:

i. *The first-order neighbours:* are cells in the first-tier around cell *A*.

ii. *The second-order neighbours:* are cells in the second-tier around cell *A*.

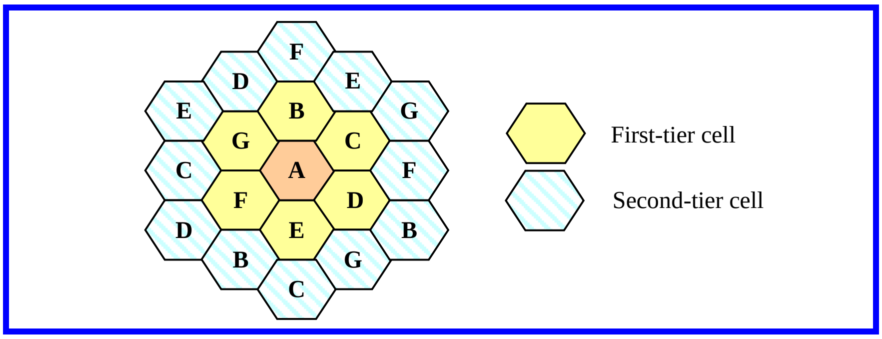


Figure (3.4). First and second tier cells.

The group of cells that forms first and second tiers around a central cell, which are shown in Figure (3.4), is called frequency reuse pattern. Depending on the previous definitions we assume that the usage of channel *k* in cell *A* imposes the following constraints to a channel allocation algorithm:

i. Channel *k* cannot be allocated to other wireless terminals in cell *A* or in the first and second tiers of cells.

ii. The two adjacent channels of *k* cannot be allocated to other wireless terminals in cell *A* or in the first-tier of cells.

However, considering the second constraint, that the adjacent channels of *k* are not allowed to be used in the first-tier of cells, reduces the total number of channels that can be allotted for each cell (*f*) in Eqn. (3.2) to be calculated as:

$$f = \frac{K}{N+2} \tag{3.5}$$

Where *K* is the total number of channels allotted for the system, *N* is the reused factor (or number of cells within the pattern).

Thus, for a GSM cellular network, since $K = 124$ and $N = 7$, then *f* is approximated to either 17 or 13 according to Eqns. (3.2) and (3.6), respectively. The first one means neglecting the adjacent channels interferences with the first tier neighbours, while the second one means taking it into consideration.

3.4 Global System for Mobile Communications (GSM)

3.4.1 Introduction and history of GSM

In the early 1980s there were analogue technologies, e.g., Advanced Mobile Phone Services (AMPS) in North America, Total Access Communications System (TACS) in the UK, and Nordic Mobile Telephone (NMT) in Nordic countries. Since each country developed its own system, a number of problems originated, such as:

- The system worked only within the boundaries of each country.
- Mobile equipment manufacturers markets were limited by the operating system.

The solution was GSM, which is a digital cellular technology that was developed in 1982 by the Conference of European Posts and Telecommunications (CEPT). It is classified as a second generation (2G) mobile phone system. GSM is an open digital cellular technology used for transmitting voice and data services. GSM differs from first generation wireless systems in that it uses digital technology and both

FDMA and TDMA transmission modes.

The GSM communications is a digital cellular communications system which has rapidly gained acceptance and market share worldwide, although it was initially developed in a European context. In addition to digital transmission, GSM incorporates many advanced services and features, including Integrated Services Digital Network (ISDN) compatibility and worldwide roaming in other GSM networks. The advanced services and architecture of GSM have made it a model for future third-generation (3G) mobile systems, such as the Universal Mobile Telephone Service (UMTS), which is a 3G mobile system being developed by ETSI within the ITU's IMT-2000 framework. It will provide data speeds of up to 2 Mbps, making portable videophones a reality. However, this section will give an overview of the services offered by GSM, the system architecture, the radio transmission structure, and the signaling functional architecture.

The CEPT formed a study group called Group Special Mobile (the initial meaning of GSM). The group was to study and develop a pan-European public cellular system in the spectrum that had been allocated in the 900 MHz range. Some of the basic criteria for their proposed system were:

- good subjective speech quality
- low terminal and service cost
- support for international roaming
- ability to support handheld terminals
- support for range of new services and facilities
- spectral efficiency
- ISDN compatibility

In 1989, the responsibility for GSM was transferred to the European Telecommunication Standards Institute (ETSI), and the Phase I recommendations were published in 1990. At that time, the United Kingdom requested a specification based on GSM but for higher user densities with low-power mobile stations (MSs), and operating at 1800 MHz. The specifications for this system, called Digital Cellular System (DCS1800) were published 1991. Commercial operation of GSM networks started in mid-1991 in European countries, and moved on to the rest of the world [Car 96, Gue 97, Pri 97, Web 5].

3.4.2 Services provided by GSM

GSM was designed having interoperability with ISDN in mind, and the services provided by GSM are a subset of the standard ISDN services. Speech is the most basic, and most important, tele-service provided by GSM. In addition, various data services are supported, with user bit rates up to 9600 bps. Specially equipped GSM terminals can connect with Public Switched Telephone Network (PSTN), ISDN, Packet Switched and Circuit Switched Public Data Networks, through several possible methods, using synchronous or asynchronous transmission. Also supported are Group 3 facsimile service, videotex, and teletex. Other GSM services include a cell broadcast service, where messages such as traffic reports, are broadcast to users in particular cells.

A service unique to GSM, the Short Message Service (SMS), allows users to send and receive point-to-point alphanumeric messages up to a few tens of bytes. It is similar to paging services, but much more comprehensive, allowing bi-directional messages, store-and-forward delivery, and acknowledgement of successful delivery. Supplementary services enhance the set of basic tele-services. In the

Phase I specifications, supplementary services include variations of call forwarding and call barring, such as Call Forward on Busy or Barring of Outgoing International Calls. Many more supplementary services, including multiparty calls, advice of charge, call waiting, and calling line identification presentation which is offered in the Phase 2 specifications.

3.4.3 System architecture

The functional architecture of a GSM system can be broadly divided into the MS, the BS subsystem, and the network subsystem.

Each subsystem is comprised of functional entities which communicate through the various interfaces using specified protocols. Figure (3.5) and Figure (3.6) show the main components of a GSM system.

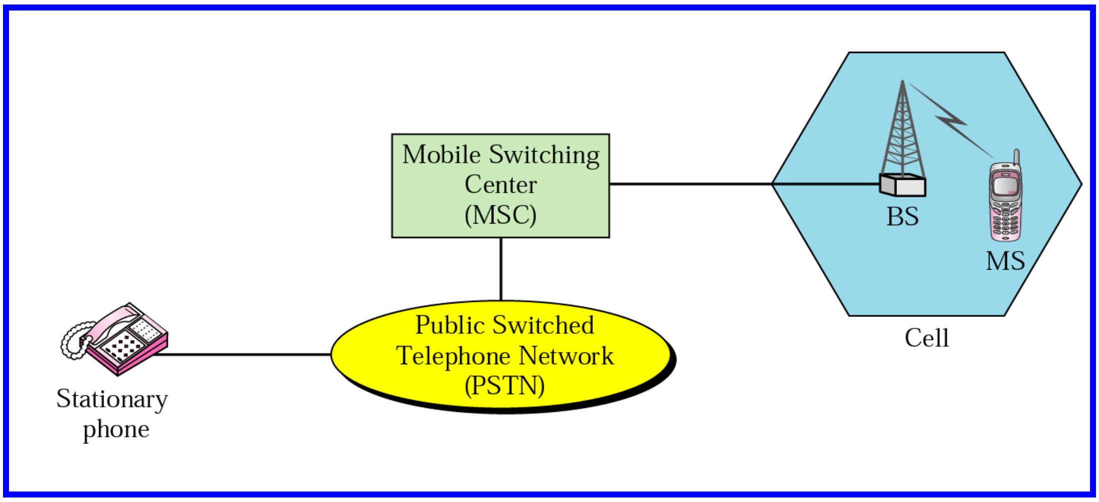


Figure (3.5). The main components of a GSM system [Sta 05].

i. *Mobile station (MS)*

The MS in GSM is really two distinct entities. The actual hardware is the mobile equipment, which is anonymous. The subscriber information, which includes a unique identifier called the International Mobile Subscriber Identity (IMSI), is stored in the Subscriber Identity Module (SIM), implemented as a smart card. By inserting the SIM card in any GSM mobile equipment, the user is able to make and receive calls at that terminal and receive other subscribed services. By decoupling subscriber information from a specific terminal, personal mobility is provided to GSM users.

ii. *Base station subsystem*

The BS Subsystem is composed of two parts, the Base Transceiver Station (BTS) and the BS Controller (BSC). The BTS houses the radio transceivers that define a cell and handles the radio (Um) interface protocols with the MS. Due to the potentially large number of BTSs, the requirements for a BTS are ruggedness, reliability, portability, and minimum cost.

The BSC manages the radio resources for one or more BTSs, across the Abis interface [Mah 07]. It manages the radio interface channels (setup, teardown, frequency hopping, etc.) as well as handovers.

iii. *Network subsystem*

The central component of the network subsystem is the Mobile Switching Center (MSC). It acts like a normal switching node of the PSTN or ISDN, and in addition provides all the functionality needed to handle a mobile subscriber, including registration, authentication, location updating, inter-MSC handovers, and call routing to a roaming subscriber.

These services are provided in conjunction with four intelligent databases, which together with the MSC form the network subsystem. The MSC also provides the connection to the public fixed networks.

The Home Location Register (HLR) contains all the administrative information of each subscriber registered in the corresponding GSM network, along with the current location of the subscriber. The location assists in routing incoming calls to the mobile, and is typically the SS7 address of the visited MSC. There is logically one HLR per GSM network, although it may be implemented as a distributed database.

The Visitor Location Register (VLR) contains selected administrative information from the HLR, necessary for call control and provision of the subscribed services, for each mobile currently located in the geographical area controlled by the VLR. Although the VLR can be implemented as an independent unit, to date all manufacturers of switching equipment implement the VLR together with the MSC, so that the geographical area controlled by the MSC corresponds to that controlled by the VLR.

The proximity of the VLR information to the MSC speeds up access to information that the MSC requires during a call.

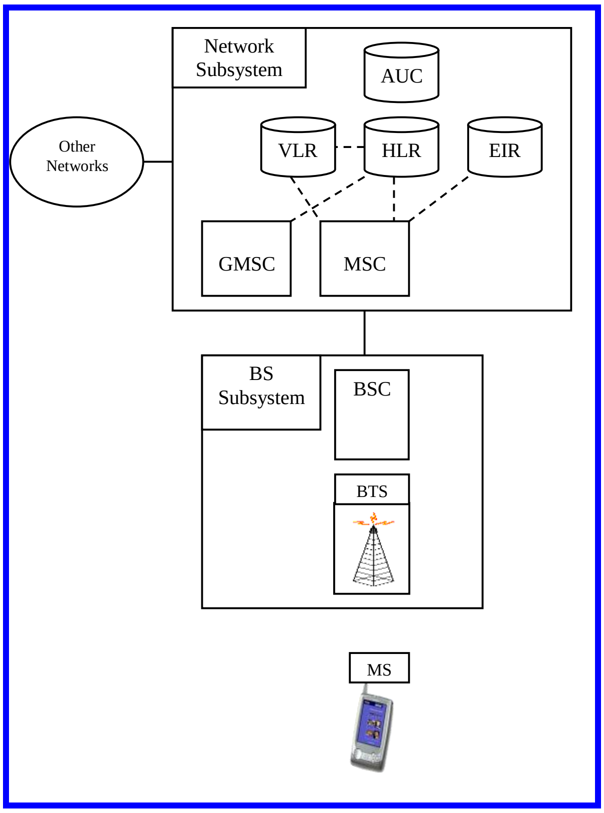


Figure (3.6). The functional components of a GSM system [Leu 07]

The other two registers are used for authentication and security purposes. The Equipment Identity Register (EIR) is a database that contains a list of all valid mobile equipment on the network, where each mobile equipment is identified by its International Mobile Equipment Identity (IMEI). An IMEI is marked as invalid if it has been reported stolen or is not type approved. The Authentication Center (AuC) is a protected database that stores a copy of the secret key stored in each subscriber's SIM card, used for authentication and ciphering on the radio channel.

3.4.4 GSM frequency bands

GSM frequency bands or frequency ranges are the radio spectrum frequencies designated by the ITU for the operation of the GSM system for mobile phones. There are 14 frequency bands defined in 3GPP TS 45.005, which succeeded 3GPP TS 05.05. In what follows, a brief description is given for the most widely used and most promising GSM bands (e.g., GSM-900, GSM-1800, GSM-850, GSM-1900, and GSM-450). But first in Table (3.2), the GSM bands are summarized. The table gives the bands, the frequency range for the up and the down links, the frequency spectrum, the carrier spacing, then number of channels, and the channel numbers. The number of channels is given by dividing the frequency spectrum by the carrier spacing.

Table (3.2)
GSM frequency bands

#	System	Band	Frequency Range		Freq. Spec. (MHz)	Carr. Spac. (KHz)	No. of Channels	Channel Number
			Uplink (MHz)	Downlink (MHz)				
1	T-GSM 380	380	380.2-389.8	390.2 - 399.8	9.6	200	47	Dynamic
2	T-GSM 410	410	410.2-419.8	420.2 - 429.8	9.6	200	47	Dynamic
3	GSM 450	450	450.4-457.6	460.4-467.6	7.2	200	35	259-293
4	GSM 480	480	478.8-486.0	488.8-496.0	7.2	200	35	306-340
5	GSM 710	710	698.0-716.0	728.0 - 746.0	18	200	89	Dynamic
6	GSM 750	750	747.0-762.0	777.0-792.0	15	200	74	438-511
7	T-GSM 810	810	806.0-821.0	851.0-866.0	15	200	74	Dynamic
8	GSM 850	850	824.0-849.0	869.0-894.0	25	200	124	128-251
9	P-GSM 900	900	890.0-915.0	935.0-960.0	25	200	124	1-124
10	E-GSM 900	900	880.0-915.0	925.0-960.0	35	200	174	975-1023 0-124
11	R-GSM 900	900	876.0-915.0	921.0-960.0	39	200	194	955-1023 0-124
12	T-GSM 900	900	870.4-876.0	915.4-921.0	6	200	29	Dynamic

13	DCS 1800	1800	1710.0-1785.0	1805.0-1880.0	75	200	374	512-885
14	PCS 1900	1900	1850.0-1910.0	1930.0-1990.0	60	200	299	512-810

- P-GSM, Standard or primary GSM 900 Band
- E-GSM, Extended GSM 900 Band (includes Standard GSM 900 band)
- R-GSM, Railways GSM 900 Band (includes Standard and Extended GSM 900 band)
- T-GSM, GSM Trunking (digital trunked radio system suitable for the land-based trunk radio)

GSM-900 and GSM-1800

GSM-900 and GSM-1800 are used in most parts of the world.

- GSM-900 uses 890-915 MHz to send information from the MS to the BS (uplink) and 935-960 MHz for the other direction (downlink), providing 124 channels (channel numbers 1 to 124) spaced at 200 kHz. Duplex spacing of 45 MHz is used.

  In some countries the GSM-900 band has been extended to cover a larger frequency range. This extended GSM (E-GSM), uses frequency range 880-915 MHz (uplink) and 925-960 MHz (downlink), adding 50 channels (channel numbers 975 to 1023 and 0) to the original GSM-900 band. The GSM specifications also describe railways GSM (GSM-R), which uses frequency range 876-915 MHz (uplink) and 921-960 MHz (downlink). Channel numbers 955 to 1023. GSM-R provides additional channels and specialized services for use by railway personnel.

All these variants are included in the GSM-900 specification.

- GSM-1800 uses 1710-1785 MHz to send information from the MS to the BS (uplink) and 1805-1880 MHz for the other direction (downlink), providing 374 channels (channel numbers 512 to 885). Duplex spacing is 95 MHz.

GSM-850

GSM-850 is used in the United States, Canada, and many other countries in the Americas. GSM-850 is also sometimes called GSM-800 because this frequency range was known as the "800MHz Band" when it was first allocated for AMPS usage in the United States in 1983.

- GSM-850 uses 824-849 MHz to send information from the MS to the Base Transceiver Station (uplink) and 869-894 MHz for the other direction (downlink). Channel numbers 128 to 251.

GSM-1900

GSM-1900 is used in the United States, Canada, and many other countries in the Americas.

- GSM-1900 uses 1850-1910 MHz to send information from the MS to the BS (uplink) and 1930-1990 MHz for the other direction

- (downlink). Channel numbers 512 to 810.

PCS is an abbreviation for Personal Communications Service and merely represents the original name in North America for the 1900 MHz band.

GSM-450

Another less common GSM version is GSM-450. It uses the same frequency as and can co-exist with old analog Nordic Mobile Telephone (NMT) systems. NMT is a first generation (1G) mobile phone system which was primarily used in Nordic countries, Eastern Europe and Russia prior to the introduction of GSM. It operates in either 450.4-457.6 MHz paired with 460.4-467.6 MHz (channel numbers 259 to 293), or 478.8-486 MHz paired with 488.8-496 MHz (channel numbers 306 to 340).

3.4.5 Radio transmission aspects

GSM is a circuit-switched system that operates in the 900 MHz and 1800 MHz bands in Europe and the 1900 MHz and 850 MHz bands in the US. GSM system has 124 pairs of simplex channels.

Each simplex channel is 200 kHz wide and supports 8 separate connections on it, using TDMA. Each currently active station is assigned one time slot on one channel pair. Theoretically, 992 channels can be supported in each cell, but many of them are not available, to avoid frequency conflicts with neighbouring cells or used for setup and control purposes.

In Figure (3.7), the 8 shaded time slots all belong to the same connection, four of them in each direction. Transmitting and receiving does not happen in the same time slot because the GSM radios cannot transmit and receive at the same time and it takes time to switch from one to the other. If the MS assigned to 890.4/935.4 MHz and time slot 2 wanted to transmit to the BS, it would use the lower four shaded slots (and the ones following them in time), putting some data in each slot until all the data had been sent [Sta 05, Tan 03].

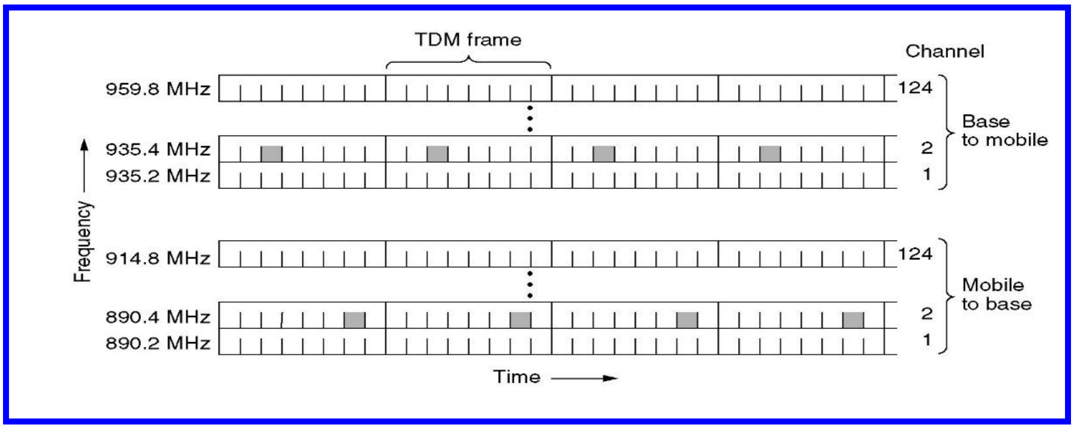


Figure (3.7). GSM uses 124 channels, each has eight-slot TDM system [Tan 03].

The TDMA slots shown in Figure (3.7) are part of a complex framing hierarchy. Each TDMA slot has a specific structure, and groups of TDMA slots form multiframes, also with a specific structure. A simplified version of this hierarchy is shown in Figure (3.8).

Here we can see that each TDMA slot consists of a 148-bit data frame that occupies the channel for 577 µsec (including a 30-µsec guard time after each slot). Each data frame starts and ends with three 0 bits, for frame delineation purposes. It also contains two 57-bit information fields; each one having a control bit that indicates whether the following information field is for voice or data. Between the information fields is a 26-bit Sync (training) field that is used by the receiver to synchronize to the sender's frame boundaries [Tan 03].

A data frame is transmitted in 547 µsec, but a transmitter is only allowed to send one data frame every 4.615 msec, since it is sharing the channel with seven other stations. The gross rate of each channel is 270,833 bps, divided among eight users.

As can be seen from Figure (3.8), eight data frames make up a TDMA frame and 26 TDMA frames make up a 120-msec multiframe. Of the 26 TDMA frames in a multiframe, slot 12 is used for control and slot 25 is reserved for future use, so only 24 are available for user traffic.

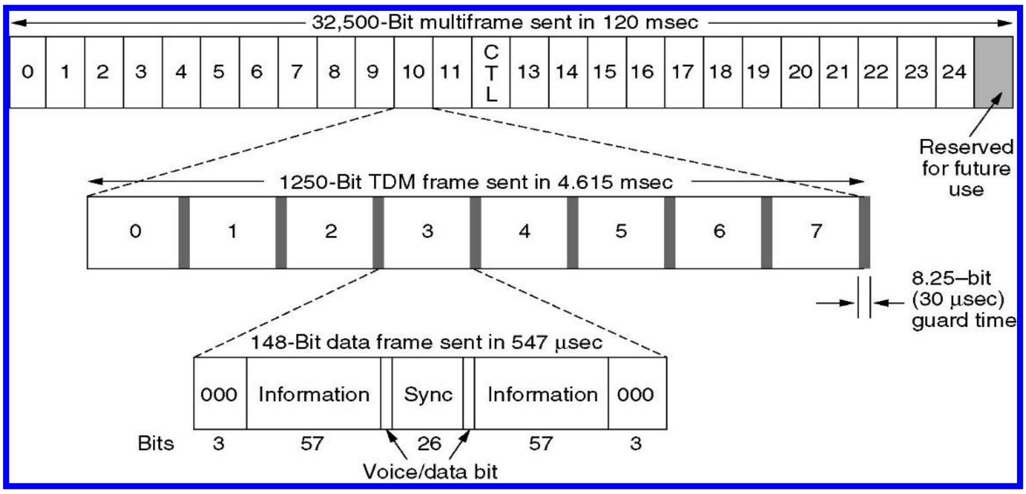


Figure (3.8). A portion of the GSM framing structure [Tan 03].

However, in addition to the 26-slot multiframe that is shown in Figure (3.8), a 51-slot multiframe (not shown) is also used. Some of these slots are used to hold several control channels used to manage the system. The broadcast control channel is a continuous stream of output from the BS containing the BS's identity and the channel status. All MSs monitor their signal strength to see when they have moved into a new cell.

The dedicated control channel is used for location updating, registration, and call setup. In particular, each BS maintains a database of MSs currently under its jurisdiction. Information needed to maintain this database is sent on the dedicated control channel.

Finally, there is the common control channel, which is split up into three logical subchannels. The first of these subchannels is the paging channel, which the BS uses to announce incoming calls. Each MS monitors it continuously to watch for calls it should answer. The second is the random access channel, which allows users to request a slot on the dedicated control channel. If two requests collide, they are garbled and have to be retried later. Using the dedicated control channel slot, the station can set up a call. The assigned slot is announced on the third subchannel, the access grant channel.

There is a basic distinction between dedicated and idle modes, which arises from on-demand channel allocation due to spectrum scarcity. Dedicated, or traffic, channels provide a bi-directional point-to-point transmission link to a mobile subscriber. Full-rate Traffic Channels (TCH/F) and half-rate Traffic Channels (TCH/H) are allocated together with a low bit-rate Slow Associated Control Channel (SACCH), which typically transmits measurements needed for handover decisions.

There are also eighth-rate traffic channels, also called Stand-alone Dedicated Control Channels (SDCCH), which are used primarily for transmitting location updating information.

In addition, a TCH slot can be pre-empted for signaling, in which case it is called a Fast Associated Control Channel (FACCH), which can be either full-rate or half-rate. TCHs are defined within a 26-frame multiframe. Common channels can be accessed both by idle mode mobiles, in order to change to dedicated mode, and by dedicated mode mobiles, to monitor surrounding BSs for handover information. The common channels, which are defined within a 51-frame multiframe, include:

- *Broadcast Control Channel (BCCH)*: Continually broadcasts, on the downlink, information including BS identity, frequency allocations, and frequency hopping sequences.

- *Frequency Correction Channel (FCCH)* and *Synchronization Channel (SCH)*: Used to synchronize the mobile to the time slot structure of a cell by defining the beginning of a TDMA frame.

- *Random Access Channel (RACH)*: Slotted Aloha channel used by the MS to request access to the network.

- *Paging Channel (PCH)*: Used to alert the MS of incoming call.
- *Access Grant Channel (AGCH)*: Used to allocate an SDCCH to a MS for signaling (in order to obtain a dedicated channel), following a request on the RACH.

3.4.6 Network aspects

Radio transmission forms the lowest functional layer in GSM. In any telecommunication system, signaling is required to coordinate the necessarily distributed functional entities of the network. The transfer of signaling information in GSM follows the layered OSI model. On top of the physical layer described above is the data link layer (DLL) providing error-free transmission between adjacent entities, based on the ISDN's Link Access Protocol - Channel D (LAPD) for the Um and Abis interfaces, and on SS7's Message Transfer Protocol (MTP) for the other interfaces. The functional layers above the DLL are responsible for Radio Resource Management (RRM), Mobility Management (MM) and Call Management (CM) [Nav 02, Web1].

The Um is an air interface used for exchanges between a MS and a BSS. The Abis is a Base Station Subsystem internal interface linking the BSC and a BTS, and it has not been standardised. The Abis interface allows control of the radio equipment and radio frequency allocation in the BTS. The A interface is between the BSS and the MSC. The A interface manages the allocation of suitable radio resources to the MSs and mobility management.

The RRM functional layer is responsible for providing a reliable radio link between the MS and the network infrastructure. This includes the establishment and allocation of radio channels on the Um interface, as well as the establishment of A interface links to the MSC. The handover procedures, an essential element of cellular systems, are managed at this layer, which involves the MS, the BS subsystem, and, to a lesser degree, the MSC. Several protocols are used between the different network elements to provide RRM functionality.

The MM functional layer assumes a reliable RRM connection, and is responsible for location management and security. Location management involves the procedures and signaling for location updating, so that the mobile’s current location is stored at the HLR, allowing incoming calls to be properly routed. Security involves the authentication of the mobile, to prevent unauthorized access to the network, as well as the encryption of all radio link traffic. The protocols in the MM layer involve the SIM, MSC, VLR, and the HLR, as well as the AuC (which is closely tied with the HLR). The machines in the network subsystem exchange signaling information through the Mobile Application Part (MAP), which is built on top of SS7.

The CM functional layer is divided into three sublayers. The Call Control (CC) sublayer manages call routing, establishment, maintenance, and release, and is closely related to ISDN call control. The idea is for CC to be as independent as possible from the underlying specifics of the mobile network.

Another sublayer is Supplementary Services, which manages the implementation of the various supplementary services, and also allows users to access and modify their service subscription. The final sublayer is the Short Message Service (SMS) layer, which handles the routing and delivery of short messages, both from and to the mobile subscriber. Figure (3.9) illustrates the GSM layers and interfaces.

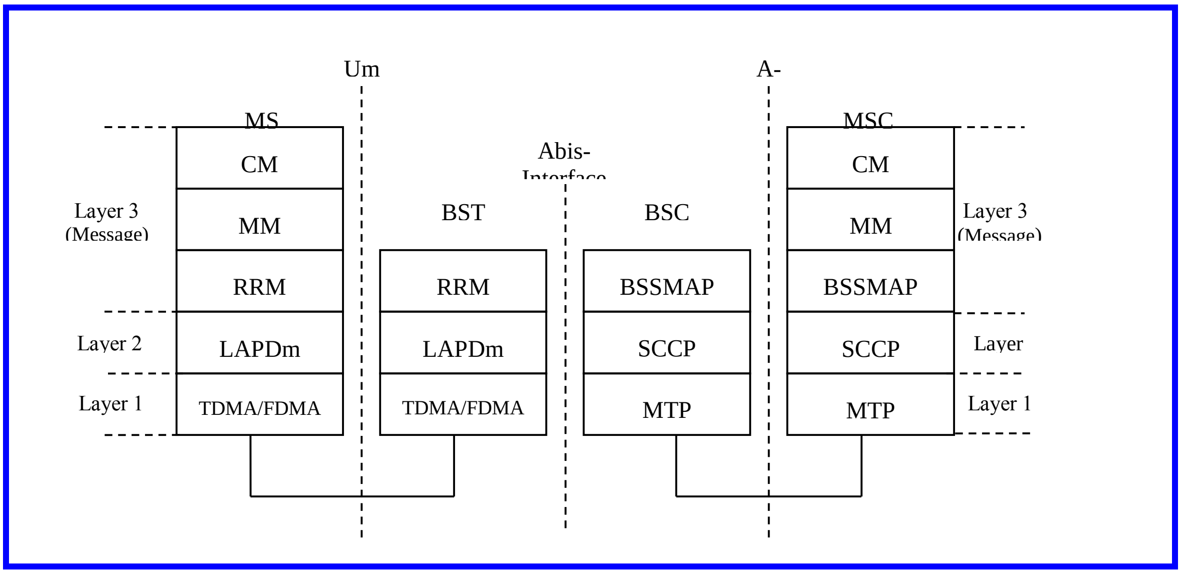


Figure (3.9). The GSM layers and interfaces [Leu 07, Sta 05].

Chapter 4
Implementation of the ABT and the AWC Algorithms

4.1 Introduction

A number of techniques have been used to combat impairments in rapidly varying radio channels and to obtain high spectral efficiencies in cellular systems. Some of those are channel coding and interleaving, adaptive modulation, transmitter/receiver antenna diversity, spectrum spreading, and dynamic channel allocation (DCA) [Gar 06, Lee 07, Ras 07]. In this work we are concerned with DCA, which is an important issue in cellular communication networks (CCNs) that requires efficient and reliable solutions [Kou 07, Per 07, Pic 07].

The DCA approach can be formulated as a distributed constraints satisfaction problem (DCSP); therefore, all techniques that are developed to solve DCSP can be used to solve DCA. Although, there are many algorithms to solve DCSP, here we present the concept, methodology, characteristics, and implementation of two algorithms, these are [Jun 04, Fal 04, Yok 00]:

i. The asynchronous backtracking (ABT) algorithm

ii. The asynchronous weak-commitment (AWC) algorithm

In order to validate our implementation, evaluate and analyze the performance of these two algorithms, they are used to solve a standard distributed artificial intelligent (DAI) problem that can be formulated as DCSP. It is the *N*-Queen puzzle [Sil 06, Yok 00].

The performance of the algorithms is compared in terms of two parameters, these are: (i) the average number of cycles ($\bar{c}$), and (ii) the success ratio (S_R).

The rest of this chapter is organized as follows. Section 4.2 presents an introduction to and application of DAI. In addition, it summarizes the main features of the applications or problems for which DAI can be applied. The definition, concept, and formalism of the constraint satisfaction problem (CSP) and the DCSP are given in Sections 4.3 and 4.4, respectively. Section 4.5 discusses the DCSP methodology. The ABT and the AWC algorithms are described in Section 4.6. In Section 4.7, the implementation and performance of these two algorithms are validated, analyzed, and compared by solving the *N*-queens puzzle for different number of queens (*N*) and constraints. In addition, in this section, the objectives of the tests and the results obtained are presented and discussed. Finally, in Section 4.8, a description is given for two mechanisms that can be used to implement the above two algorithms for DCA in CCNs, namely, the on-demand and the token-based mechanisms.

4.2 Distributed Artificial Intelligence (DAI)

4.2.1 Definitions

DAI is a sub field of artificial intelligence (AI) that is concerned with interaction, especially coordination among artificial automated agents [Yok 98]. Since distributed computing environments are spreading very rapidly due to the advances in hardware, software, and networking technologies, there are pressing needs for DAI techniques. DAI is dedicated to the development of distributed solutions for complex problems regarded as requiring intelligence.

DAI techniques are extremely useful and widely used in solving and studying multi-agent systems (MASs). Today, MASs are a very active area of research and are beginning to be extensively used in many commercial and industrial applications. Also, the exponential growth of the Internet and its ability to connect people and machines from all over the world is making the arrival of MASs an inescapable consequence of such high degree of interconnectivity.

DAI is an extension of ideas derived from AI that applies to MASs. Instead of one centralized and usually very large application that encodes the complete intelligence of the system, a number of relatively small systems, or agents, are involved in a cooperative effort to resolve a problem. This does not imply that the large system is merely divided into smaller pieces, but there are several centralized applications, each capable of addressing a certain aspect of a problem and can be tied together by a communication system. It would allow for exchange of their viewpoints and coming up with strategies to make progress or to combine the results into a solution. This kind of problem solving is called distributed problem solving (DPS) [Wei 99, Wil 01].

DAI aims to construct systems composed of multiple problem solving entities, which interact with one another to enhance their performance. With this “divide and conquer” approach the scope of each component is limited; meaning the complexity of the computation is lower, thus enabling the processing elements to be simpler and more reliable. This increased demand on software systems has also coincided with important advances in hardware technology, meaning it is now economically feasible and technically viable to connect together large numbers of powerful, yet inexpensive, processing units that execute asynchronously [Jen 93].

Most DAI research has concentrated on developing communities in which both control and data are distributed. Distributed control means that individuals have a degree of autonomy in generating new actions and in deciding which tasks to do next. When designing such systems it is important to ensure that agents spend the bulk of their time engaged in solving the domain level problems for which they were built, rather than in communication and coordination activities [Jen 93, Wei 99].

The main reason to deal with DAI is that MASs have the capacity to play a key role in current and future computer science and its application. Modern computing platforms and information environments are distributed, large, open, and heterogeneous. Computers are no longer stand-alone systems, but have become tightly connected both with each other and their users. The increasing complexity of computer and information systems goes together with an increasing complexity of their applications.

These often exceed the level of conventional, centralized computing because they require, for instance, the processing of huge amounts of data, or of data that arises at geographically distinct locations [Wei 06].

4.2.2 Applications of DAI

Many existing and potential industrial and commercial applications for DAI and MASs are described in the literature, examples of such applications are:

1. Real-time monitoring and management of telecommunication networks, where agents are responsible, for example, for call forwarding and signal switching and transmission.

2. Electronic commerce and electronic markets, where "buyer" and "seller" agents purchase and sell goods on behalf of their users.

3. Modeling and optimization of in-house, in-town, national or world-wide transportation systems, where agents represent, for example, the transportation vehicles, the goods, or the customers to be transported.

4. Information handling in information environments, like the Internet, where multiple agents are responsible for information filtering and gathering.

5. Improving the flow of urban or air traffic, where agents are responsible for appropriately interpreting data arising at different sensor stations.

6. Automated meeting scheduling, where agents act on behalf of their users to fix meeting details like location, time, and agenda.

7. Optimization of industrial manufacturing and production processes like shop-floor scheduling or supply chain management, where agents represent, different work cells or whole enterprises.

8. Analysis of business processes within or between enterprises, where agents represent the people or the distinct departments involved in these processes in different stages and at different levels.

9. Electronic entertainment and interactive virtual reality-based computer games, where animated agents equipped with different characters play against each other or against humans.
10. Design and re-engineering of information- and control-flow patterns in large-scale natural, technical, and hybrid organizations, where agents represent the entities responsible for these patterns.
11. Investigation of social aspects of intelligence and simulation of complex social phenomena, such as: the evolution of roles, norms, and organizational structures, where agents take on the role of the members of the natural societies under consideration.

What these applications have in common is that they show one or several of the following features [Wei 06]:

1. Inherent distribution: they are inherently distributed in the sense that the data and information to be processed.

2. Arise at geographically different locations and different times.

3. Structured into clusters whose access and use requires familiarity with different ontologies and languages (semantic distribution) and/or are structured into clusters whose access and use requires different perceptual, effectual, and cognitive capabilities (functional distribution).

4. Inherent complexity: they are inherently complex in the sense that they are too large to be solved by a single, centralized system because of limitations available at a given level of hardware or software technology.

4.3 Constraint Satisfaction Problem (CSP)

CSP is one of the most successful problem solving paradigms in AI. It has found numerous applications in almost all areas of AI. The most common applications of CSP are in configuration, planning, scheduling and resource allocation. In addition, it forms a basis for significant software industries [Yok 00].

The main strength of CSP derives from its ability to flexibly combine a set of constraints with its own specific solution sets. A constraint satisfaction achieves this combination through the use of consistency and search techniques. Consistency is used to eliminate many possibilities for each local constraint, and search is used to find a consistent solution within the space of possibilities and still preserves consistency [Fal 05].

CSPs have been defined for a centralized architecture. A constraint network is defined by a triple (X, D, C) where $X=\{X_1 \ldots X_N\}$ is a set of N variables, $D=\{D(X_1), \ldots, D(X_N)\}$ is the set of their respective finite domains, and C is a set of constraints specifying the acceptable value combinations for variables. The CSP involves finding values for the variables that satisfy all the constraints [Che 01].

The constraints here involve two variables only, namely binary constraints. General CPSs may involve constraints of any arity, but since network communication is only pairwise, the focus in this work will be on this subclass of problems which is binary. A constraint between X_i and X_j is denoted by C_{ij}. Any two variables are said to be constrained if and only if there is a conflict between their values [Fal 04, Yok 00].

For example, in the 8-queens problem, which will be discussed and solved in this chapter, it is obvious that only one queen can be placed in each row. Therefore, we can formalize this problem as a CSP, in which there are 8 variables X_1, X_2, . . ., X_8, each of which corresponds to a position of a queen in each row. The domain of a variable is {1, 2,..., 8}. A constraint between any 2 variables is that they must not be in the same row, column or in the same diagonal. A solution is a combination of values of these variables [Yok 00].

Another typical example problem is a graph coloring problem. The objective of a graph-coloring problem is to paint nodes in a graph so that any two nodes connected by a link do not have the same color. Each node has a finite number of possible colors. This problem can be formalized as a CSP by representing the color of each node as a variable, and the possible colors of the node as a domain of the variable. If all constraints are binary (i.e., between two variables), a CSP can be represented as a graph, in which a node represents a variable, and a link between nodes represents a constraint between the corresponding variables.

If the variables of CSP are distributed among agents, then solving a CSP in which multiple agents are distributed is called a DCSP, and can be considered as achieving coherence among the agents. Many application problems in DAI can be formalized as DCSPs, such as:

interpretation problems, assignment problems, multiagent truth maintenance tasks, and resource allocation in communication systems [Mod 01].

4.4 Distributed Constraint Satisfaction Problem (DCSP)

Cellular wireless systems, computer networks, and Internet pose themselves in a multi-agent setting where variables and/or constraints of the problem are controlled by different centralized or distributed agents.

Distributed, collaborative agents play an important role in large-scale multiagent applications such as cellular networks. Collaborative agents in such applications must coordinate their plans, resolving conflicts, if any, among their resource choices. DCSP is a major technique in multiagent coordination and conflict resolution in collaborative settings. DCSP provides rich foundation for the representation of multiagent coordination and conflict resolution, and there exist highly efficient baseline algorithms [Yok 00].

The combined problem is simply the aggregation of variables and constraints from different agents into a single problem. Each variable and constraint in the resulting DCSP is owned by one particular agent who ensures that the variable is assigned a value and constraint is satisfied. The actual search for solving the DCSP can be carried out by a central agent, but most work has focused on distributed search through message exchange among the agents. If all knowledge about the problem can be gathered into one agent, this agent could solve the problem alone using traditional centralized constraint satisfaction algorithms. However, such a centralized solution is often inadequate or even impossible.

The advantage of DCSP is that an agent only need to know about agents that own a variable it has a constraint with, but not about the entire problem, Figure (4.1) [Fal 05, Jun 04]. It is particularly applicable to problems of coordination between agents. Constraints are a good notion to express dependencies between agents' actions, and DCSP algorithms exploit this structure to localize communication among agents. Such coordination problems occur, for example, in military or transportation planning, and there are numerous application opportunities in electronic commerce. Another application area of DCSP is in networks of simple, identical agents such as sensor networks.

In DCSP, each agent is assigned one or more variables along with the constraints on their variables. The goal of each agent, from a local perspective, is to ensure that each of the constraints on its variables is satisfied. Clearly, each agent's goal is not independent of the goals of the other agents in the system, but the goals of the agents are strongly interrelated. For example, in order for one agent to satisfy its local constraints, another agent, potentially not directly related through a constraint, may have to change the value of its variable.

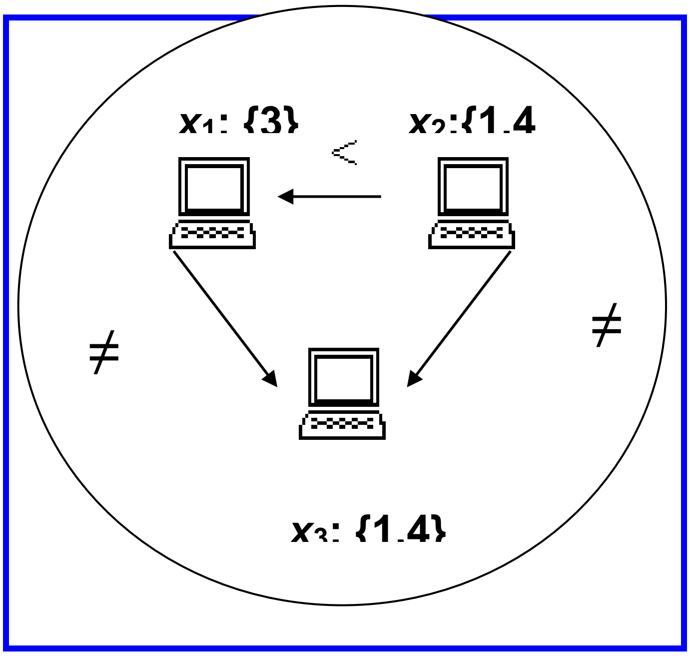


Figure (4.1). Example of constraints [Yok 98].

Formally, DCSP is a CSP where the variables, domains and constraints of the underlying network are distributed among agents. A finite variable-based distributed constraint network is defined by a

5-tuple (X, D, C, A, F), where X, D and C are as defined before. A={1, . . ., *p*} is a set of *p* agents, and F: X→A is a function that maps each variable to its agent, where each variable belongs to one agent. As in the centralized case, a solution of a DCSP is an assignment of values to variables satisfying every constraint. DCSPs are solved by the collective action of agents A, each holding a process of distributed constraint satisfaction [Bes 05].

4.5 The DCSP Methodology

In a distributed resource allocation problem in a CCN, each agent has its own tasks, and there are several ways (plans) to perform each task. Since resources are shared among agents, there exists constraints/contention between plans. The goal is to find the combination of plans that enables all the tasks to be executed simultaneously. This problem can be formalized as a DCSP by representing each task as a variable, and possible plans as variable values. Since a variety of DAI application problems can be formalized as DCSPs, therefore, distributed algorithms for solving DCSPs are considered as an important infrastructure in DAI

In a DCSP, the following communication model is assumed [Yok 00]:

i. Agents communicate by sending messages. An agent can send messages to other agents if the agent knows the addresses of the agents.

ii. The delay in delivering a message is finite, though random. For any pair of agents, messages are received in the order in which they were sent.

It must be noted that this model does not necessarily mean that the physical communication network must be fully connected (i.e., a complete graph). Unlike most parallel/distributed algorithm studies, in which the topology of the physical communication network plays an important role, the existence of a reliable underlying communication structure among the agents is assumed; and do not care about the implementation of the physical communication network. This is because the primary concern in this research is cooperation among intelligent agents, rather than solving DCSPs by certain multiprocessor architectures.

There are problems that could be solved using traditional centralized constraint satisfaction algorithms. However, such a centralized solution is often inadequate or even impossible. Here are some reasons why distributed methods may be desirable [Yok 98].

1. Cost of creating a central authority: A constraint satisfaction problem may be naturally distributed among a set of peer agents. In such cases, a central authority for solving the problem would require adding an additional element that was not present in the architecture. Examples of such systems are sensor networks, or meeting scheduling.

2. Knowledge transfer costs: In many cases, constraints arise from complex decision processes that are internal to an agent and cannot be articulated to a central authority. Examples of this range from simple meeting scheduling, where each participant has complex preferences that are hard to articulate, to coordination decisions in virtual enterprises that result from complex internal planning. A centralized solver would require such constraints to be completely articulated for all possible situations. This would entail prohibitive costs.

3. Privacy/Security concerns: Agents constraints may be strategic information that should not be revealed to competitors, or even to a central authority. This situation often arises in e-commerce and virtual enterprises. Privacy is easier to maintain in distributed solvers.

4. Robustness against failure: The failure of the centralized server can be fatal. In a distributed method, a failure of one agent can be less critical and other agents might be able to find a solution without the failed agent. Such concerns arise for example in sensor networks, but also in web-based applications where participants may leave while a constraint solving process is ongoing.

These reasons mainly have motivated significant research activities in DCSP. The field has reached a certain maturity and has developed a range of different techniques [Yok 00]. All of the above characteristics of DAI and DCSP methodologies, made an inescapable motivation to try to use them to achieve efficient and reliable DCA in CCNs.

4.6. Techniques for Solving DCSPs

There are many algorithms that have been developed to solve DCSPs. The most widely used algorithms are:

1. The asynchronous backtracking (ABT) algorithm.
2. The asynchronous weak-commitment (AWC) algorithm.

Following, is a detailed description, main features, and implementation of these two algorithms:

4.6.1 The asynchronous backtracking (ABT) algorithm

The ABT algorithm is a distributed, asynchronous search algorithm for solving DCSPs, developed by [Yok 92], and considered as the first sound and complete algorithm that is asynchronous, distributed and concurrent, in which the agents can roll up in a competitive and asynchronous way. ABT was derived from the backtracking algorithm, which is a basic, systematic search algorithm for solving CSPs. ABT allows agents to run concurrently and asynchronously. Each agent instantiates its variable and communicates the variable value to the relevant agents.

The concept of backtracking is the core of this algorithm, in ABT a value assignment to a subset of variables that satisfies all of the constraints within the subset is constructed. This value assignment is called a partial solution. A partial solution is expanded by adding new variables one by one, until it becomes a complete solution. When for one variable, no value satisfies all of the constraints with the partial solution, the value of the most recently added variable to the partial solution is changed. This operation is called backtracking. In the ABT, the priority order of variables/agents is static and determined by the alphabetical order of the variable identifiers. An agent changes its assignment if its current value assignment is not consistent with the assignments of higher priority agents [Yok 92, Yok 00, Bej 04, Sil 06, Ezz 07].

Relying on the definition of a DCSP where there are *N* agents and *N* variables. Each variable belongs to one agent, the ABT algorithm can be summarized as follows [Yok 98]:

1. ABT is executed autonomously by each agent in the network allowing agents to run concurrently and asynchronously.
2. In ABT, variables are nodes; constraints are directed links between nodes. Since each agent has exactly one variable, a node also represents an agent.

3. Each agent instantiates its variable concurrently and communicates the variable value to agents which are connected by outgoing links and each agent takes its own decisions, and no agent has to wait for decisions of others.

4. ABT computes a global consistent solution (or detects that no solution exists) in finite time; its correctness and completeness (always finds a solution if one exists, and terminates if no solution exists) is guaranteed.

5. ABT requires constraints (links) to be directed; one of the two agents involved in a constraint is assigned that constraint, and receives the other agent's value. A link is directed from the value-sending agent to the constraint-evaluating agent, which means a constraint causes a directed link between the two constrained agents, Figure (4.2).

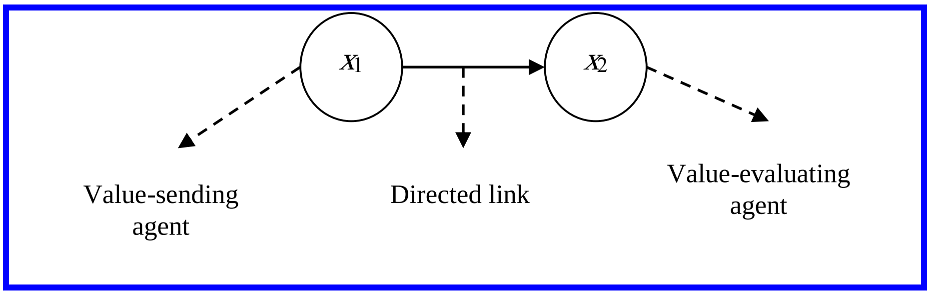


Figure (4.2). Constrained agents [Yok 98].

6. To make the network cycle-free there is a total order among agents, which is followed by the directed links.

7. Each agent keeps its own *agent_view* and nogood store.
8. For any agent X_i, the *agent_view* of X_i is the set of values that it believes to be assigned to agents connected to X_i by incoming links.
9. For any agent X_i, the *nogood* store keeps *nogoods* as justifications of inconsistent values.
10. Agents exchange assignments and *nogoods*. When agent X_i makes an assignment, it informs those agents connected to it by outgoing links by sending ok message.
11. Agent X_i always accepts new assignments, updating its *agent_view* accordingly.
12. When agent X_i receives a *nogood*, it is accepted if it is consistent with X_i's *agent_view*, otherwise it is discarded as obsolete.
13. An accepted *nogood* is added to X_i *nogood* store to justify the deletion of the value it targets.
14. When X_i cannot take any value consistent with its *agent_view*, because of the original constraints or because of the received

15. *nogoods*, new *nogood* message is generated as inconsistent subset of the *agent_view*, and is sent to the closest agent involved, causing backtracking.

16. The process terminates when achieving quiescence, meaning that a solution has been found, or when the empty *nogood* is generated, meaning that the problem is unsolvable.

Figure (4.3) describes the ABT procedures executed by agent X_i for receiving two kinds of messages and the related functions.

Receiving *ok* Message procedure

```
when received (ok?, ($x_j$,$d_j$))
{
    add ($x_j$,$d_j$) to agent_view;
    check_agent_view;
}
```

Receiving *nogood* message procedure

```
when received (nogood , $x_j$ , nogood)
{
    add nogood to nogood_list;
    when ($x_k$,$d_k$) where $x_k$ is not connected  is contained in nogood
    {
        request $x_k$ to add a link from $x_k$ to $x_i$;
        add ($x_k$ , $d_k$) to agent_view ;
    }
    old_value = current_value;
    check_agent_view;
    when old_value = current_value
    {
        send (ok?, ($x_j$,current_value)) to $x_j$;
    }
}
```

Check *agent_view* function

```
when agent_view and current_value are not consistent
{
    if (no value in D is consistent with agent_view) then
    {
        backtrack;
    }
    else
    {
        select $d_j$ belongs D where agent_view and $d_i$ are consistent;
        current_value ← $d_j$ ;
```

```
            send (ok?, ($x_i$,$d_j$)) to outgoing links;
      }
}
                              Backtrack function
nogoods = {V | V = inconsistent subset of agent_view}
when an empty set is an element of nogoods
{
      broadcast to other agents that there is no solution;
      terminate this algorithm;
}
for each V belongs nogoods
{
      select ($x_j$, $d_j$) where $x_j$ has the lowest priority in V;
      send (nogood, $x_i$, V) to $x_j$;
      remove ($x_j$, $d_j$) from agent_view;
}
      check_agent_view;
 }
```

Figure (4.3) ABT procedures and functions [Yok 98].

Avoiding infinite processing loops

If agents change their values again and again and never reach a stable state, they are in an infinite processing loop. An infinite processing loop can occur if there exists a value changing loop of agents, such as if a change in X_1 causes X_2 to change, then this change in X_2 causes X_3 to change, which then causes X_1 to change again, and so on. In the network representation, such a loop is represented by a cycle of directed links. One way to avoid cycles in a network is to use a total order relationship among nodes. If each node has a unique identifier, we can define a priority order among agents by using the alphabetical order of these identifiers (the preceding agent in the alphabetical order has higher priority). If a link is directed by using this priority order (from the higher priority agent to the lower priority agent), there will be no cycle in the network. This means that for each constraint, the lower priority agent will be an evaluator, and the higher priority agent will send an ok message to the evaluator. Furthermore, if a *nogood* is found, a *nogood* message is sent to the lowest priority agent in the *agent_view* [Yok 98].

From the preceding we could see that each agent has to know only the identifiers of an agent with which it must establish a constraint in order to direct the constraint, but not all the agents in the network, this means each agent must establish a directed link from itself to lower priority agents if it has a constraint with, and agents communicate their location values with each other, by sending ok messages from each agent to all lower priority agents.

Also agent's own assignment is consistent with the *agent_view* if all constraints the agent evaluates are true under the value assignments described in the *agent_view* and if all communicated *nogoods* are not compatible with the *agent_view*, if an agent's own assignment is not consistent with the *agent_view* and the agent is not able to find any consistent value with the *agent_view*; a subset of an *agent_view* is called a *nogood* that caused the problem is defined and the agent will select the least priority agent from its *agent_view* and will send to it a *nogood* message, then the least priority agent will be removed from its *agent_view*. The algorithm will terminate when a solution is found or when an empty *nogood* element set is found, so there is no solution.

Figure (4.4) shows an example for the execution of the ABT algorithm, where agents communicate via ok and *nogood* messages.

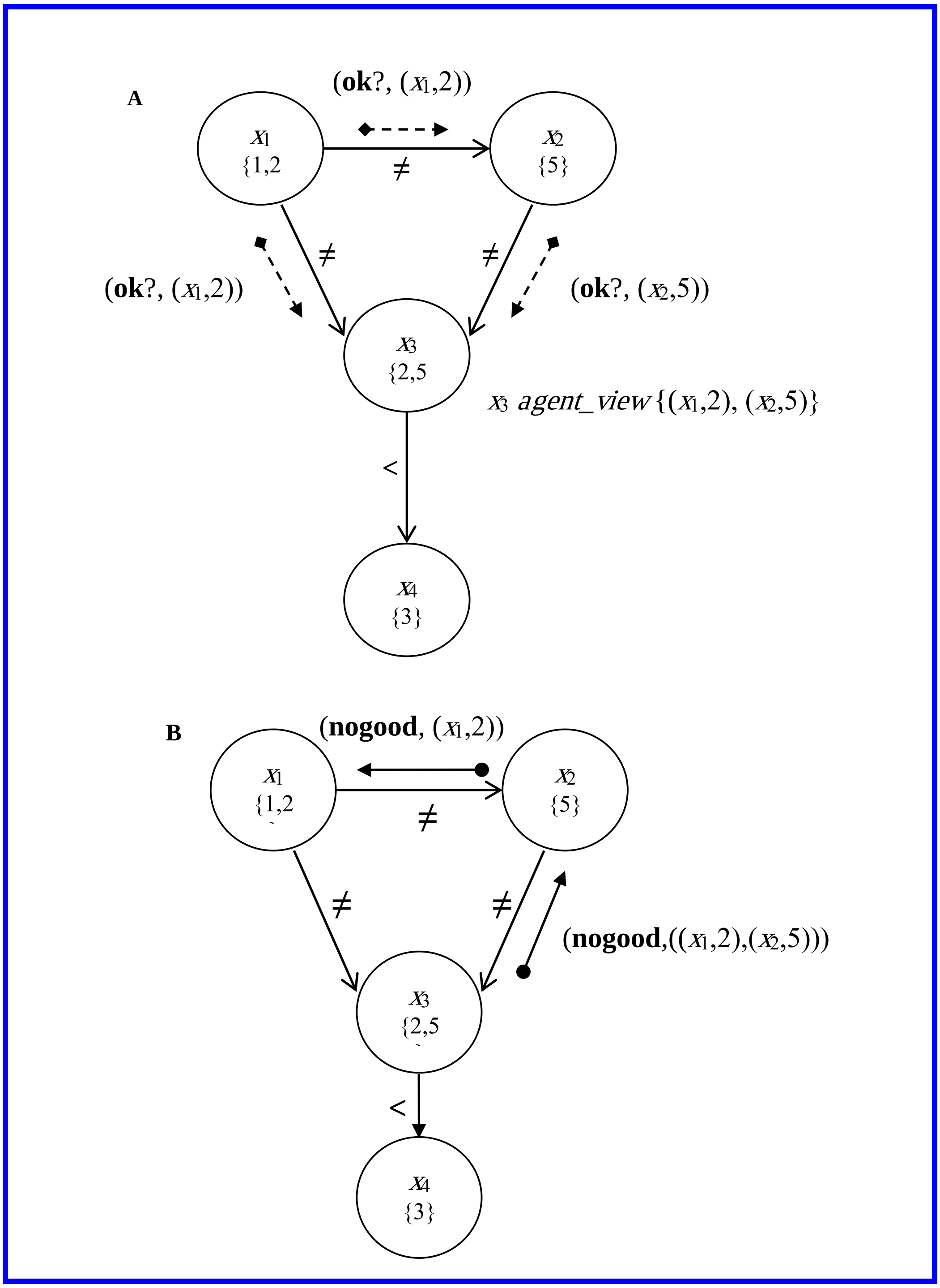


Figure (4.4). Example illustrates the execution of the ABT algorithm.

4.6.2 The asynchronous weak-commitment (AWC) algorithm

The AWC algorithm was proposed and developed by Makoto Yokoo et. al in 1995 for solving DCSP [Yok 95]. It is based upon the ABT algorithm and inspired by the weak-commitment algorithm [Yok 94]. Two main keys represent the strength of this algorithm; it uses the min-conflict heuristic as a value ordering heuristic to reduce the risk of making bad decisions, and it abandons the partial solution and restarts the search process if there is no consistent value with the partial solution, which means AWC weakly commits to a partial solution that constructed from bad decisions.

In AWC agents asynchronously assign values to their variables and communicate the values to neighboring agents with shared binary constraints, the priority order is dynamically changed using the communicated priority values. If the current value is not consistent because some constraint with variables of higher priority agents is not satisfied, the agent changes its value so that the value is consistent, and also the value minimizes the number of constraint violations with variables of lower priority agents. But when it cannot find a consistent value, it sends *nogood* messages to other agents, and increments its priority value. If it has already sent an identical *nogood*, it will not change its priority value but will wait for the next message.

A solution is a value assignment in which every variable is consistent. The AWC algorithm records the abandoned partial solutions as new constraints, and avoids creating the same partial solution that has been created and abandoned before [Yok 95, Yok 00, Jun 04, Sae 05, Bej 04, Sil 06].

In the AWC algorithm, all variables have temporal initial values. A consistent partial solution is constructed for a subset of variables, and this partial solution is extended by adding variables one by one until a complete solution is found. When a variable is added to the partial solution, its tentative initial value is revised so that the new value satisfies all the constraints between the variables included in the partial solution, and satisfies as many constraints as possible between variables that are not included in the partial solution, this value ordering heuristic is called the min-conflict heuristic which has a great effect on finding a solution with small number of cycles as possible [Fal 04].

When there exists no value for one variable that satisfies all the constraints between the variables included in the partial solution, this algorithm abandons the whole partial solution, and starts constructing a new partial solution from scratch, using the current value assignment as new tentative initial values. This algorithm records the abandoned partial solutions as new constraints, and avoids creating the same partial solution that has been created and abandoned before. Therefore, the completeness of the algorithm is guaranteed [Yok 98].

Features of the AWC algorithm

The main two features of the AWC algorithm can be described as follows [Che 01]:

1. Min-conflict: when selecting a variable value, if there are multiple values consistent with the *agent_view*, the agent prefers the value that minimizes the number of constraint violations with variables of lower priority agents.

2. Weak-commitment: when an agent cannot find a value consistent with the current *agent_view*, it increases its priority value to be the maximum of neighbors. The mechanism of dynamically changing priority whenever a new *nogood* created; enables agents weakly commit to the partial solution, which

3. means it abandons the partial solution and restarts the search process if there exists no consistent value with the partial solution. By increasing a priority value in this way, a wrong variable value of a high priority agent can be revised without performing exhaustive search by lower priority agents, which is the main characteristic of the AWC algorithm.

Since agents act concurrently and asynchronously, and no agent has exact information about the partial solution, furthermore, multiple agents may try to restart the search process simultaneously, so for establishing the priority order and changing the priority values, the following rules will control the process [Yok 98]:

- For each agent, a nonnegative integer value representing the priority order of the agent is defined; this value is called the priority value.

- The order is defined such that any agent with a larger priority value has higher priority.

- If the priority values of multiple agents are the same, the order is determined by the alphabetical order of the identifiers.

- For each agent, the initial priority value is 0.

- If there exists no consistent value for agent X_i, the priority value of X_i is changed to k +1, where k is the largest priority value of the related agents.

The differences between the procedures performed by the ABT and AWC algorithms are as follows [Yok 98]:

1. In the ABT, each agent sends its variable value only to related lower priority agents, while in the AWC algorithm each agent sends its variable value to both lower and higher priority agents connected by constraints.

2. In the AWC algorithm the priority value, as well as the current value assignment, is communicated through the ok message, but in ABT only the current value is communicated.

3. In the AWC algorithm the priority order is determined using the communicated priority values, which means the priority is dynamically changed, while the priority is fixed in the ABT algorithm.

4. If an agent in AWC has to change its value and there are many consistent values, then the agent will choose the value

5. that minimizes the number of constraint violations with variables of lower priority agents. In ABT if the agent has many consistent values, then the agent will choose a value randomly.

6. In the AWC algorithm if an agent has already sent an identical *nogood*, the agent will not send another *nogood* message, but in ABT the agent may resend the same *nogood* message if the case is same.

The AWC algorithm works as follows:

1. There exist *N* agents.

2. Each agent has to know only the identifiers of an agent with which it must establish a constraint in order to direct the constraint.

3. The priority value is initially zero for all agents, but it is dynamically changed during the execution, if two agents have the same priority value the alphabetical order of identifiers will determine who is greater (i.e., X_1 priority is higher than X_2).

4. Each agent must establish a directed link from itself to lower priority agents if it has a constraint with.

5. Each agent has a set of values from the agents that are connected by incoming links. These values constitute the agent's *agent_view* which has the same definition as in the ABT algorithm.

6. Agents communicate their location values and priority values with constrained agents, by sending ok messages for both lower and higher priority agents, the lower and higher priority agents are called neighbours.

7. After that, the agents wait for and respond to messages.

8. If an ok message is received by some agent by an incoming link, the evaluating agent adds the message sending agent index, priority, and value to his neighbors, and if the agent who sent the message has a higher priority, then it will be added to the evaluating agent's *agent_view*, and then checks whether its own value assignment is consistent with its *agent_view*.

9. The agent’s own assignment is consistent with the *agent_view*, if all constraints the agent evaluates are true

10. under the value assignments described in the *agent_view*.

11. If an agent's own assignment is not consistent with the *agent_view*, the agent will try to change the current value so that it will be consistent with its *agent_view* and also the value minimizes the number of constraint violations with lower priority agents, and then send ok message to his neighbors.

12. If an agent's own assignment is not consistent with the *agent_view* and the agent is unable to find any consistent value, then a subset of the *agent_view* which caused the problem is defined and is called a *nogood*, after that the agent will check if it has already sent an identical *nogood* message to this subset, if yes it will not change its priority nor send *nogood* message but will wait for another message.

13. If the agent has not sent an identical *nogood* message to the same subset, then the agent will increase its priority value to be the maximum priority value of all agents connected to it by constraints plus one, then it will change the current value to another value that minimizes the number of constraint violations with lower priority agents, and then send ok message to his neighbours.

14. If a *nogood* message is received by some agent by an outgoing link, the agent will check if its current value is consistent with its *agent_view*, depending on the result the agent will respond as described before.

15. The algorithm will terminate when a solution is found or when there is an empty *nogood* element set is found, so there is no solution.

Figure (4.5) summarizes the AWC procedures when agents receive *ok* and *nogood* messages, and other related functions.

Receiving *ok* procedure

```
when received (ok?, ($x_j$,$d_j$, priority))
{
    add ($x_j$,$d_j$,priority) to agent_view;
    check_agent_view;
}
```

Receiving *nogood* procedure

```
when received (nogood , $x_j$ , nogood)
{
    add nogood to nogood_list;
    when ($x_k$,$d_k$,priority) where $x_k$ is not in neighbors is contained in nogood
    {
        add $x_k$ to neighbors, add ($x_k$ ,$d_k$,priority) to agent_view;
    }
    check_agent_view;
}
```

Check_*agent_view* function

```
when agent_view and current_value are not consistent
{
    if (no value in D is consistent with agent_view) then
    {
        backtrack;
    }
    else
    {
        select $d_i$ belongs D where agent_view and $d_i$ are consistent and $d_i$ minimizes the
            number of constraint violations with lower priority agents;
        current_value = $d_i$;
        send (ok?, ($x_i$, $d_i$, current_priority)) to neighbors;
    }
}
                              Backtrack procedure
when an empty set is an element of nogoods
{
    broadcast to other agents that there is no solution, terminate this algorithm;
}
when no element of nogoods is included in nogood_sent
{
    for each V belongs nogoods
    {
        add V to nogood_sent;
        for each ($x_j$, $d_j$, $p_j$) in V
        {
            send (nogood, $x_i$, V) to $x_j$;
        }
        $P_{max}$ = max($x_j$,$d_j$, $p_j$) belongs agent_view ($p_j$);
        current_priority = 1 + $P_{max}$;
        select $d_i$ belongs D where $d_i$ minimizes the number of constraint violations with lower
            priority agents;
        current_value = $d_i$;
        send (ok?, ($x_i$, $d_i$, current_priority)) to neighbors;
    }
}
```

Figure (4.5). AWC procedures and functions [Yok 98].

4.7 Validation of the ABT and the AWC Algorithms

4.7.1 N-Queens puzzle

The *N*-Queens puzzle is derived from the chess game, where the *N*-Queens board is composed of *N* columns and *N* rows, there will be *N* queens distributed on the board, in such a way that each queen will allocate one of the *N* rows as shown in Figure (4.6), the puzzle will be solved if and only if the *N* queens are distributed on the *N*x*N* board in such a way that there is only one queen at each row, column, and diagonal, so that there is no queen threatens any other queen.

In general, queens will stand for agents (nodes) in a cellular communication network; agents will stand for a base station, which is responsible for controlling all mobiles within a cell. The directed links between the queens will represent constraints between agents, constraints stand for the legal usage of the available channels, legal channel reuse, or for any resource there is a constraint for using it.

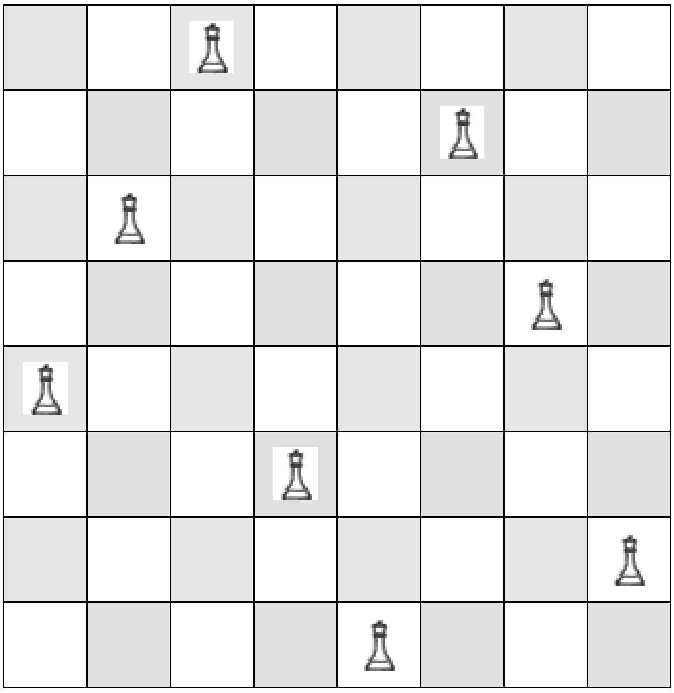


Figure (4.6). 8-Queens board [Yok 98].

4.7.2 Implementation

The two algorithms are coded using Borland Java Builder9 Enterprise edition, since Java has many advantages and programming abilities that could be used to implement distributed and constrained environments, such as the multithreading technique, which will be used to represent agents that may work concurrently and asynchronously, each agent will be represented by a single thread that works independently from other threads, each thread will process its own data and data from the thread environment, and then responds without waiting other threads to finish.

The performance of the two algorithms is evaluated using a discrete event simulation clock, where each agent maintains its own simulated clock. An agent's time is incremented by one simulated time unit, whenever it performs one cycle of computation. One cycle consists of:

1. Reading all incoming messages
2. Performing local computation
3. Sending messages.

We assume that a message issued at time t is available to the recipient at time t+1.

4.7.3 Performance measures

The performance of the ABT and the AWC algorithms, for solving the *N*-Queens puzzle, are analyzed and compared in terms of two parameters, these are: (i) the average number of cycles ($\bar{c}$), and (ii) the success ratio (S_R). The definition of the above two parameters is given below.

i. The average number of cycles ($\bar{c}$)

It is defined as the number of cycles required to establish a valid queens distribution that satisfies all constrains. It also gives indication on the average processing time, where the average processing time is directly proportional to $\bar{c}$, and its absolute value depends on the processor speed. One cycle corresponds to a series of agent actions, in which an agent recognizes the state of the world (the value assignments of other agents), then decides its response to that state (its own value assignment), and communicates its decisions.

The calculation of $\bar{c}$ is straightforward. For each *N*, *T* trials are performed; each trial has different random initial queens distribution. For each trial, in order to conduct the experiments within a reasonable amount of time, we set the limit for the maximum number of cycles to establish a valid solution (queens distribution that satisfies all constraints) at C_{max}. If the algorithm exceeds C_{max} before a valid solution is established, the trial is terminated. Otherwise, if the algorithm establishes a valid solution, the trial is terminated, and the number of cycles (c_i) acquired to establish a valid solution and its square are accumulated to be used to calculate $\bar{c}$ and its associated standard deviation (□). In addition, for each successful trial, one is added to a counter S that is initially set to zero. Thus, $\bar{c}$ is calculated as:

$$\bar{C} = \frac{\sum_{i=1}^{S} c_i}{S} \qquad (4.1)$$

The associated standard deviation (□) is expressed as:

$$\sigma = \sqrt{\frac{1}{S}\sum_{i=1}^{S}\left(c_i - \bar{C}\right)^2} \qquad (4.2)$$

ii. The success ratio (S_R)

It represents the probability of achieving a valid solution before the number of cycles exceeds C_{max}. S_R is obtained by dividing the number of successful trials (S) by the total number of trails (T). It can be mathematically expressed as:

$$S_R = \frac{S}{T} \qquad (4.4)$$

4.7.4 Simulation objectives, results, and discussion

In order to evaluate the performance of the ABT and the AWC algorithms for solving distributed and constrained problems; the two algorithms are used to solve the *N*-Queens puzzle. The main objectives of this test are:

- Illustrate the ability of the two algorithms to solve the *N*-Queens puzzle in a distributed environment.
- Compare the performance of the two algorithms to find out which algorithm will outperform the other.
- Figure out the complications that may arise when applying the two algorithms to such constrained and distributed environments.

Thus, implementation of the two algorithms to solve the *N*-Queens puzzle is just a tool to test their capabilities to solve distributed and constrained problems.

The results obtained for solving the *N*-Queens puzzle using the ABT and the AWC algorithms are summarized in Table (4.1).

Table (4.1)

Comparison of the performance of the ABT and the AWC algorithms for solving the *N*-Queens Puzzle.

N	C_{max}	T	ABT Algorithm			AWC Algorithm			K
			$\overline{C}$	σ	S_R	$\overline{C}$	σ	S_R	
25	200	1000	84	62	0.55	48	33	100	0.57
50	400	500	132	100	0.36	47	28	100	0.36
75	500	250	168	131	0.22	52	29	100	0.31
100	600	200	208	133	0.13	48	32	100	0.23
125	750	200	264	184	0.08	52	34	100	0.20

Where $K = \overline{C}$ (AWC) / $\overline{C}$ (ABT).

The results obtained demonstrate that as *N* increases, the average number of cycles needed to solve the problem increases too. This is because, as *N* increases, the number of exchanged messages will increase and the number of agents that frequently change their locations will increase too; this leads to more number of cycles to solve the problem.

For all values of *N*, $\overline{C}$ for the AWC algorithm is less than that for the ABT algorithm by a factor of 2 to 5. In addition, for the AWC algorithm, $\overline{C}$ is slightly increased with *N*, e.g., for N=50 and *N*=125, $\overline{C}$ are 47 and 52, respectively.

Most important, it is clear that the AWC algorithm can always establish a valid solution ($S_R = 1$) regardless of the initial distribution and number of queens. For the ABT algorithm, S_R is less than 1, i.e., it is not always possible to establish a valid distribution for the queens within the certain number of cycles (C_{max}). However, a solution may be established if C_{max} increases.

Comparing the results of the two algorithms we could see that the AWC algorithm can solve problems that cannot be solved within a reasonable amount of computation time by ABT, also the AWC will always need less number of cycles to solve the problem whatever the value of N. The efficiency of AWC came from two key techniques; first the priority order is dynamic, second the use of min-conflict heuristic.

When the priority order is static, which is the case in ABT, the efficiency of the algorithm is highly dependent on the selection of initial values, and the distribution of required cycles is quite large. When the initial values of higher priority agents are good, the solution can easily be found. If some of these values are bad, however, an exhaustive search is required to revise these values; this tends to make the number of required cycles exceed the limit. On the other hand, in the AWC, the initial values are less critical, and a solution can be found even if the initial values are far from the final solution, since the variable values gradually come close to the final solution.

If the priority is static, the misjudgements (bad value selections) of agents with higher priority are fatal to all agents. On the other hand, by changing the priority order dynamically and selecting values cooperatively, the misjudgements of specific agents do not have fatal effects, since bad decisions are weeded out, and only good decisions survive. These results are intuitively natural, since they imply that a flexible agent organization performs better than a static and rigid organization.

The min-conflict heuristic in AWC is used in the cases where there are multiple valid values that could be assigned to some agent; the agent prefers the value that minimizes the number of constraint violations with variables of lower priority agents, this will lead to decrease the number of cycles needed to solve the problem since each agent will try always to leave some valid values for the lower priority agents if that is possible.

4.8.CCNs Implementation Mechanisms

The resource allocation in CCNs can be formulated as DCSP, so that it can be solved using either the ABT algorithm or the AWC algorithm. To formulate the resource allocation in cellular communication networks as DCSP, we assume the following:

i. An agent stands for a BS (cell), which is responsible for controlling all mobile phones within the cell.

ii. Variables stand for the channels allocated for each cell

iii. Constraints stand for the legal usage of the available channels and the legal channel reuse to eliminate any radio interferences.

Obviously, two mechanisms can be adopted in implementing the above two algorithms for dynamic resource allocation in cellular mobile networks, these are:

1. On-demand mechanism

   In this mechanism, each cell collaboratively updates its resources (allocate new channels as needed and as available). Cells instantly and simultaneously exchange information on the allocated frequencies as depicted by the algorithm.

Therefore, it is clear from the description of the two algorithms that they involve a huge data communications between cells and may cause a lot of interruption during the network operation. This mechanism is very much similar to the implementation suggested by M. Yokoo et. al [Yok 00].

2. Token-base mechanism

   In order to practically implement these two algorithms efficiently, for dynamic resource allocation, in wireless cellular networks with minimum communication overheads, minimum system interruption, and not affecting the system operation, a token-based mechanism is introduced.

In a token-based mechanism, each pattern (a group of collaborative cells) will have a token that circulates between the cells carrying information about the channels that are currently allocated for each cell. Also, only a cell that has the token can update its resources. So that the data communication is minimized and the process of resource update does not interrupt the network operation. Since, the token is continuously circulating between collaborative cells; if possible, a cell updates its resources if it is equal to or less than a certain threshold value. Thus, each cell can update its resources independently and as long as the resource is available.

However, in this mechanism, a cell needs to have a mechanism to initiate a resource update, such as when the cell resources below a certain threshold value, otherwise it just bypasses the token to minimize delay.

In this work the AWC algorithm has been modified to be applied in CCNs, this modification has been divided into three main functions, *Receive_Token*, *Hold_Token*, and *Min_Interference_Heuristic*, the pseudo code for these functions is given in Figures (4.7), also the flowchart for applying the modified AWC using the token-based mechanism on CCNs is shown in Figure (4.8).

```
Function Receive_Token ()
{
        For each cell identifier in Token do
             Update allocated channels for the cell; End do;
    If no. of allocated channels <= Threshold_value
       Hold_Token;
        Else
         {  If there are unused channels
             { Dislocate most of the unused channels such that no. of
               allocated channels still>Threshold_vlue;
              Register the allocated channels in the Token; }
      Bypass Token; }
}
```

```
Function Hold_Token ()
{
    {X}: set of unused channels by neighbors and self;
        If number of elements in {X} > number of needed channels
          { {Y}: set of channels;
               {Y}← Min_Interference_Heuristic(X);
                 Allocate channels in {Y};
                 Register (Y + previously allocated channels) in the Token; }
            Else if {X} not empty
          { Allocate channels in {X};
            Register (X + previously allocated channels) in the Token; }
    Bypass Token;
 }
```

```
Function Min_Interference_Heuristic (X)
{
  Ch = channel, least = 1000000, Value=0;
     While (no. of allocated channels <= Threshold_Vlaue) do
        { For each channel in X do
            { For each neighbor in my neighbors do
              { If neighbor$_i$ can allocate X$_i$
            Value = Value + 1;
          If no. of channels allocated for neighbor$_i$ <= it's Threshold_Value
         Value = Value + 1;
       } end do;
       If Value<least
         { Ch = Xi; Least = Value; }
                 } end do;
                  Allocate Ch;
                  Remove Ch from X;
                Least = 1000000;
              Value =0;
            } end do;
          Register allocated channels in the Token;
       Bypass Token;
    }
```

Figure (4.7), The modified AWC algorithm pseudo code.

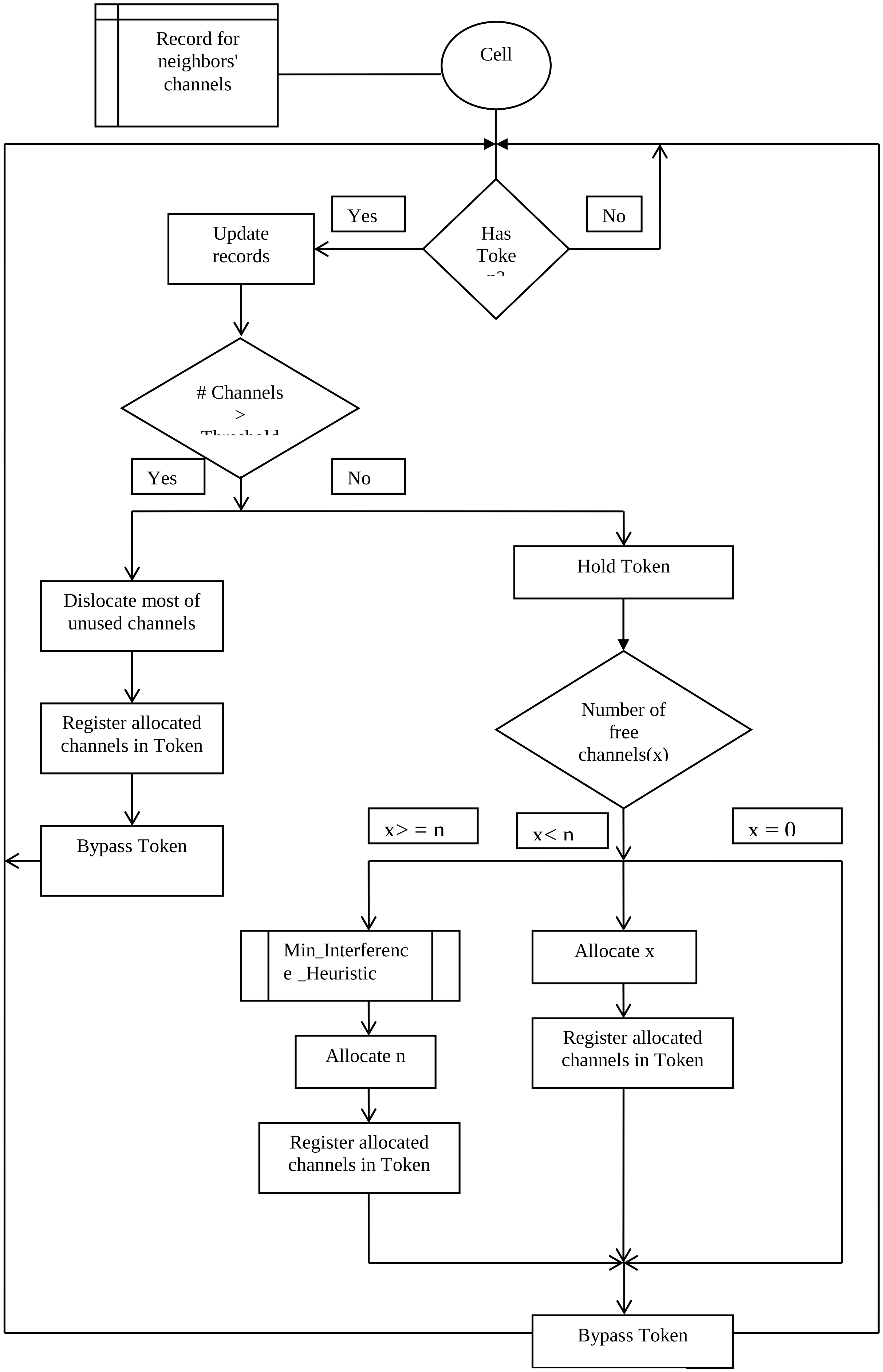


Figure (4.8). Modified AWC in CCNs Flowchart.

Chapter 5
Simulations and Performance Analysis

5.1 .Introduction

In chapter 4, the results obtained for the solution of the *N*-Queen puzzle demonstrate that the performance of the AWC algorithm overwhelms the performance of the ABT algorithm in two aspects:

(i) It can establish a solution regardless of the size and type of the problem (i.e., number of variables, number of constraints, and variables initial distribution).

(ii) It can establish this solution in less number of cycles (processing time) as compared to the processing time required by the ABT algorithm.

Thus, we shall devote this chapter to investigate and evaluate the performance of the new DCA scheme that is based on the AWC algorithm in solving more realistic problems.

In order to investigate and analyze the performance of the new scheme, a discrete event simulator was developed. It was used to simulate two different scenarios that evaluate the effect of increasing demand for channels (traffic load in calls/sec) in GSM system. These scenarios represent two different network operation environments, these are:

1. Scenario #1. Uniform initial and traffic loads.
2. Scenario #2. Uniform initial load and nonuniform traffic load.

The rest of this chapter is organized as follows. Section 5.2 describes the network operation environment. The metrics that are used in measuring and comparing the performance of the proposed scheme are introduced in Section 5.3. Finally, in Section 5.4, the results obtained for the above scenarios are presented in tables and graphs. In addition, in Section 5.4, the results are discussed.

5.2. Network Operation Environment

Initially, each cell within the network is assumed to have a number of frequencies (channels) that are allocated to satisfy the minimum interference constraints described in Section 3.3. This is also called initial load distribution, where all channels are considered as busy channels. Each cell is assumed to have a record of the channels that are allocated to all other cells within the frequency reuse pattern, through the use of the token-based scheme that is described in Section 4.6.

There are two types of initial load distribution: uniform and non-uniform. In a uniform initial load distribution, equal number of channels is allocated to all cells, while in a non-uniform initial load distribution; different numbers of channels are allocated to the different cells. However, in both cases, the initial load that is considered is less than the maximum possible uniform load.

Call arrivals which we refer to as traffic load in calls/sec can be categorized as uniform or non-uniform. In a uniform traffic load calls arrive at the same rate at all cells within the network. While, in a non-uniform traffic load calls arrive to cells at different rates, where each cell may have its own traffic load. However, it is assumed that the traffic load of a cell remains constant during the whole simulation.

In addition, a network can be described as operating in a steady-state or unsteady-state condition. The steady-state network operation condition means that all calls arrive at the BS of a cell are served or a channel is instantly allocated to all arrived calls. In other word, the number of concluded calls (calls normally terminated by network users) is equal to the number of arrived calls. This steady-state network operation condition may be disturbed due to variation in the traffic load in one or more cells to stimulate the network to operate in an unsteady-state condition.

The unsteady-state network operation condition means that the number of calls which arrive at the BS is more than the number of concluded calls; therefore, extra resources (channels) are required to be allocated for the disturbed cell(s). Otherwise, the number of dropped (failed or denied) calls is increased.

5.3 Performance Measures

There are a number of parameters that can be used to evaluate the performance of the new scheme, in providing a DCA solution in CCNs that satisfies all network operating constraints. In this work, the following parameters are considered:

i. Number of successfully allocated channels ($C_{a,i}$) which is defined as the number of channels that are successfully allocated for the i^{th} cell during network operation, to satisfy specific values and distributions of initial and traffic loads.

ii. Number of failed channels ($C_{f,i}$) which is defined as the number of channels that are failed to be allocated (i.e., calls are dropped), for the i^{th} cell during network operation, to satisfy specific values and distributions of initial and traffic loads.

iii. Efficiency (E_i). Which we define as the number of channels that are successfully allocated ($C_{a,i}$) divided by the total number of channels that are requested during the simulation period ($C_{t,i}$), where $C_{t,i} = C_{a,i} + C_{f,i}$. Thus, we can express E_i as:

$$E_i = \frac{C_{a,i}}{C_{t,i}} = \frac{C_{a,i}}{C_{a,i} + C_{f,i}} \tag{5.1}$$

Since, the scheme is usually used to compute these parameters for a number of neighbouring cells; it has been found that it is more indicative to evaluate the performance of the algorithm in terms of the average values of the above parameters. The average value of $C_{a,i}$ is denoted by $\overline{C_a}$ and it represents the average value of channels that is successfully allocated over a network of k cells, for a certain period of operation and network environment. The average value ($\overline{C_a}$) and its associated standard deviation ($\square_a$) can be calculated by the following well-known equations:

$$\overline{C_a} = \frac{\sum_{i=1}^{k} C_{a,i}}{k} \tag{5.2}$$

$$\sigma_a = \sqrt{\frac{1}{k}\sum_{i=1}^{k}\left(C_{a,i} - \overline{C_a}\right)^2} \tag{5.3}$$

Similarly, the average value of $C_{f,i}$ is denoted by $\overline{C_f}$ and it represents the average number of channels that failed to be allocated over a network of *k* cells, for a certain period of operation and network environment. Similarly the average value ($\overline{C_f}$) and its associated standard deviation (□$_f$) is given by the following well-known equations:

$$\overline{C_f} = \frac{\sum_{i=1}^{k} C_{f,i}}{k} \tag{5.4}$$

$$\sigma_f = \sqrt{\frac{1}{k}\sum_{i=1}^{k}\left(C_{f,i} - \overline{C_f}\right)^2} \tag{5.5}$$

Finally, the average value of E_i is denoted by $\overline{E}$ and it represents the average efficiency at which the algorithm successfully allocate channels over a network of *k* cells for a certain period of operation and network environment. Similarly the average efficiency ($\overline{E}$) and its associated standard deviation (□$_e$) is given by the following well-known equations:

$$\overline{E} = \frac{\sum_{i=1}^{k} E_i}{k} \tag{5.6}$$

$$\sigma_e = \sqrt{\frac{1}{k}\sum_{i=1}^{k}\left(E_i - \overline{E}\right)^2} \tag{5.7}$$

We have also found that the average efficiency ($\overline{E}$) can be calculated by:

$$\overline{E} = \frac{\overline{C_a}}{\overline{C_a} + \overline{C_f}} \tag{5.8}$$

Evaluating the performance of the DCA scheme in terms of the computed average values is very useful. This is because the algorithm may perform well and allocate all channels required by a certain cell, but it may fail to allocate enough channels for another demanding cell.

In this thesis, the effects of a number of networks parameters are investigated, these include:

i. Cell traffic load (□) which represents the number of calls that arrive to the BS (calls/sec).

ii. Cell initial load (□) which represents the number of channels that are initially allocated to a cell (channel/cell).

5.4 Simulations and Results

A number of simulations are carried-out to evaluate and analyze the performance (efficiency) of the DCA scheme and consequently the

AWC algorithm in providing a DCA solution in CCNs, which satisfies all constraints within a reasonable processing time and minimum data exchange.

These simulations are grouped into two different scenarios. These two scenarios are summarized as follows:

i. Scenario #1. Uniform initial and traffic loads

   It simulates a cellular network in suburban or rural areas, where the traffic load is low and uniformally distributed.

ii. Scenario #2. Uniform initial load and nonuniform traffic load

   It simulates a cellular network in an urban downtown area, where the traffic load is centralized in shopping or business districts.

The main difference between these two scenarios is in the traffic requirements, while the infrastructure topology of the network is unchanged, as shown in Figure (5.1), and the initial load is always uniform regarding its initial value (number of channels per cell). The total number of cells *N* that are considered is 37 cells, with one central cell encircled by three tiers.

In both scenarios, the duration of calls is modelled as a random variable that varies between 20 and 360 sec with an average value of 180 sec approximately. The arrivals of calls in each cell are modelled as independent and distributed processes with various arrival rates assigned for the neighbouring cells. For the simplicity of analysis, the hand-offs is not considered. The total simulation time is taken to be 1000 cycles, and each cycle is equivalent to 1 sec.

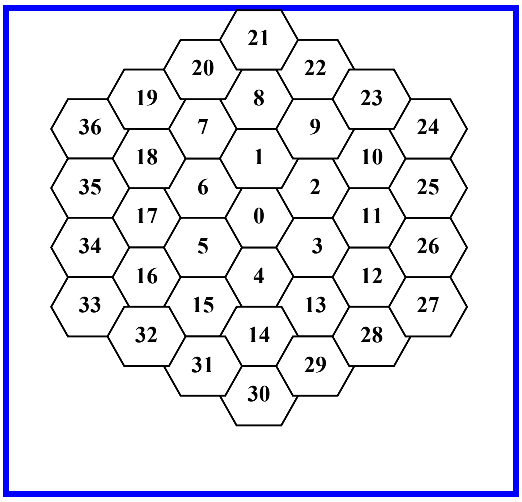


Figure (5.1). Topology of a cellular network.

In all simulations, the AWC algorithm begins with a certain initial load (□), where each cell will be allocated a specified number of channels considered as busy channels. Thus, after initialization, if the number of incoming (arrived) calls is more than the normally concluded calls, new channels need to be allocated. In other words, each cell requests channels for the new coming calls depending on the traffic load. It is important to realize that each channel can accommodate 8 calls, because GSM divides its channels into 8 time slots (TDMA). It is important to point out that the discrete event simulator has been used to study a number of scenarios of different network operation environments. But only the results for the following two scenarios are presented as they demonstrate the accuracy, efficiency, reliability, and flexibility of the proposed solution. Following, is a detailed description of these two scenarios, the results obtained, and discussion of the results are presented.

5.4.1 Scenario #1. Uniform initial and traffic loads

The first scenario simulates a network with a uniform traffic load (α = 0.2 calls/sec) in each cell, and uniform initial loads of □□□= 8 and □□□= 10 channels/cell. This scenario may represent a cellular network in a suburban or a rural area that is characterized by a uniform initial load, and relatively low and uniform traffic load.

The results obtained for $C_{a,i}$, $C_{f,i}$, and E_i using the DCA scheme are summarized in Table (5.1). It demonstrates that the scheme has an excellent performance in a suburban and rural area as it responses positively to almost all requests. It provides an efficiency of 100% (i.e., the number of dropped calls is 0) for all cells when □□= 0.2 and □□=

8. However, when □ increases to become 10, then the number of free channels are reduced which causes channel allocation failure and in some cells some calls are dropped. In addition, it can be easily recognized that $C_{a,i}$, $C_{f,i}$, and E_i for all cells are almost the same, especially when the cells initial loads are low and uniform (□ = 8). The average values ($\overline{C_a}$, $\overline{C_f}$, and $\overline{E}$) and their associated standard deviations ($\square_a$, $\square_f$, and $\square_e$) are computed using Eqns. (5.2) to (5.7) and listed in Table (5.2).

The values of $C_{a,i}$ and $C_{f,i}$ and their respective averages ($\overline{C_a}$ and $\overline{C_f}$) are shown in Figures (5.2) and (5.3), respectively, while E_i and $\overline{E}$ are shown in Figure (5.4) for both values of □ (8 and 10 channels/cell).

For this scenario, when □ = 8 channels/cell, the scheme succeeds to allocate resources to all incoming calls, and the number of calls that are denied services is zero for all cells. Thus, the scheme achieves an average allocation efficiency of 100% as all arrived calls granted services. Also, However, when □ = 10 channels/cell, the scheme still offers an excellent performance as it provides an average allocation efficiency of 96%, see Table (5.2) and Figure (5.4). When □ = 10 channels/cell, all cells are allocated the required resources (channels), except cells number 0, 5, 14, and 31, see Figure (5.3). It is important to know that this is a random process, but these results are averaged over a number of runs.

Table (5.1) - Results for Scenario #1
Values of $C_{a,i}$, $C_{f,i}$, and E_i for a uniform traffic load (α = 0.2 calls/sec) for all cells and a uniform initial load (□□= 8 and 10 Channels).

Cell	□□= 8			□□= 10		
	$C_{a,i}$	$C_{f,i}$	E_i (%)	$C_{a,i}$	$C_{f,i}$	E_i (%)
0	4	0	100	3	2	60
1	4	0	100	2	0	100
2	4	0	100	2	0	100
3	2	0	100	2	0	100
4	2	0	100	2	0	100
5	3	0	100	2	1	67
6	4	0	100	2	0	100
7	4	0	100	3	0	100
8	2	0	100	3	0	100
9	2	0	100	2	0	100
10	2	0	100	2	0	100
11	3	0	100	2	0	100
12	4	0	100	2	0	100
13	2	0	100	2	0	100
14	5	0	100	2	1	67
15	3	0	100	2	0	100
16	4	0	100	2	0	100
17	4	0	100	2	0	100
18	4	0	100	2	0	100
19	2	0	100	3	0	100
20	3	0	100	4	0	100
21	3	0	100	3	0	100
22	5	0	100	3	0	100
23	5	0	100	3	0	100
24	3	0	100	3	0	100
25	2	0	100	3	0	100
26	2	0	100	3	0	100
27	2	0	100	2	0	100
28	3	0	100	2	0	100

29	4	0	100	3	0	100
30	3	0	100	3	0	100
31	3	0	100	1	1	50
32	3	0	100	3	0	100
33	3	0	100	4	0	100
34	3	0	100	2	0	100
35	3	0	100	2	0	100
36	3	0	100	2	0	100

Table (5.2) - Results for Scenario #1

Values of $\overline{C_a}$, $\overline{C_f}$, $\overline{E}$, and their respective standard deviations (σ_a, σ_f, and σ_e) for a uniform traffic load (α = 0.2 calls/sec) and for a uniform initial load (β = 8 and 10 channels/cell).

β	$\overline{C_a}$	σ_a	$\overline{C_f}$	σ_f	$\overline{E}$ (%)	σ_e
8	3.16	0.93	0.00	0.00	100	0.00
10	2.43	0.65	0.14	0.42	96	0.13

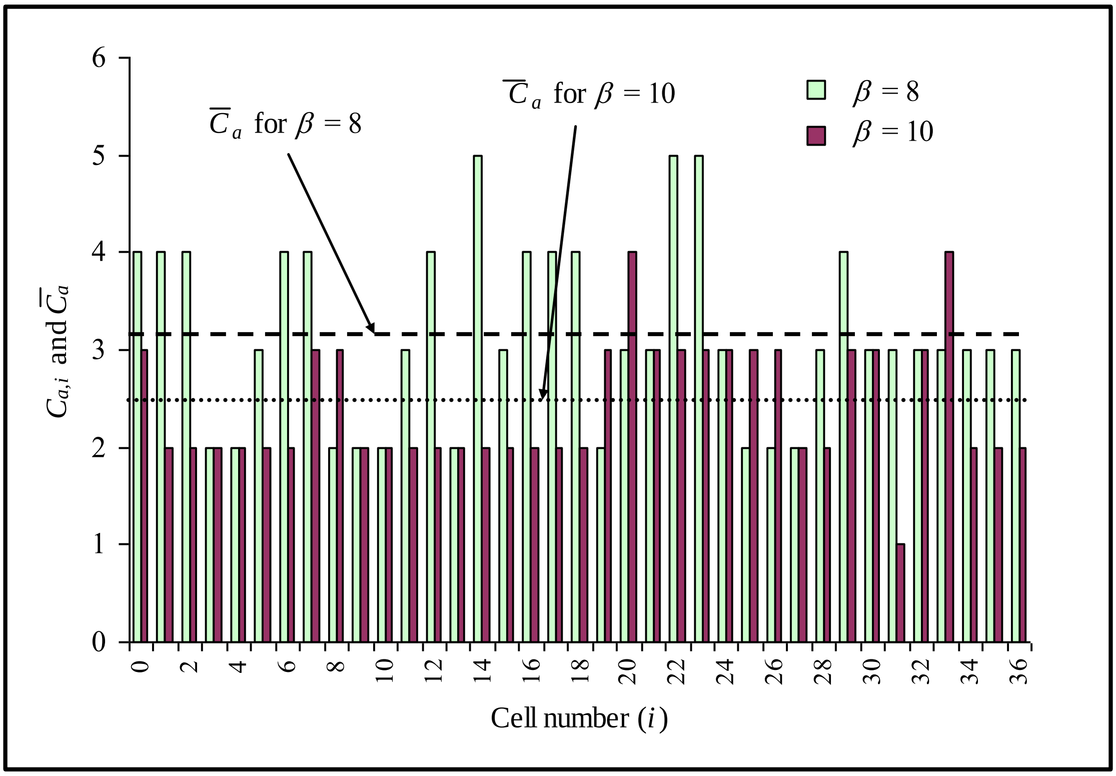


Figure (5.2). $C_{a,i}$ versus cell number (i) for Scenario #1.

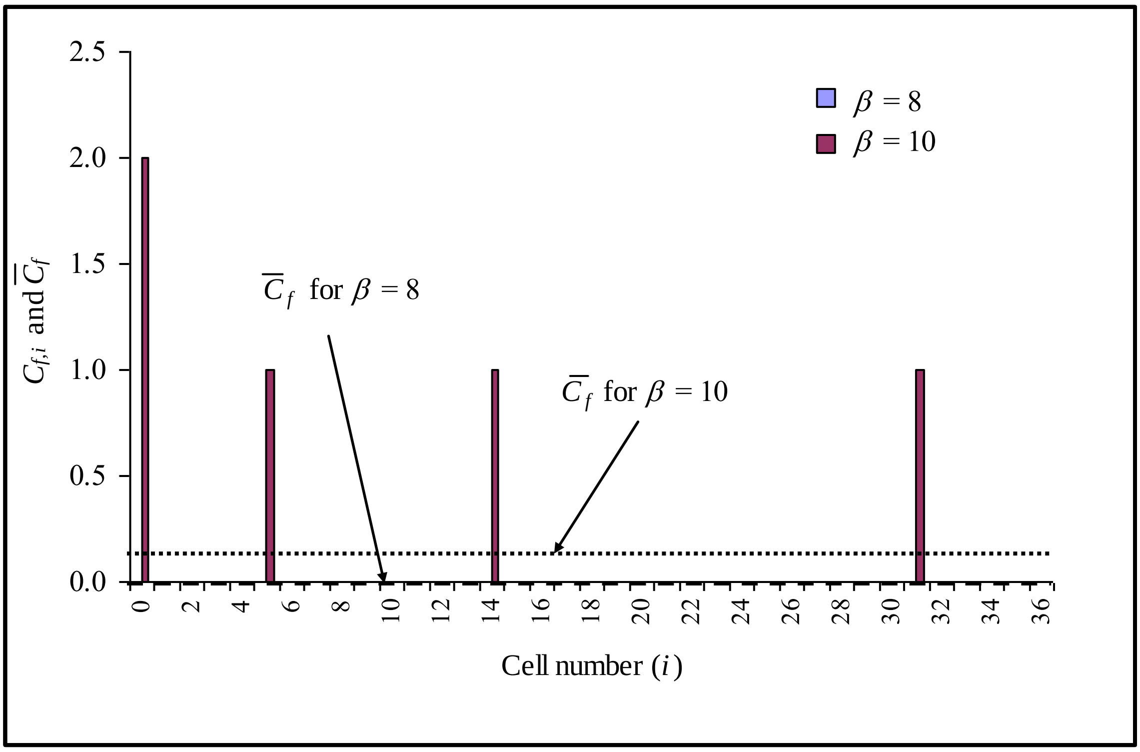


Figure (5.3). $C_{f,i}$ versus cell number (i) for Scenario #1.

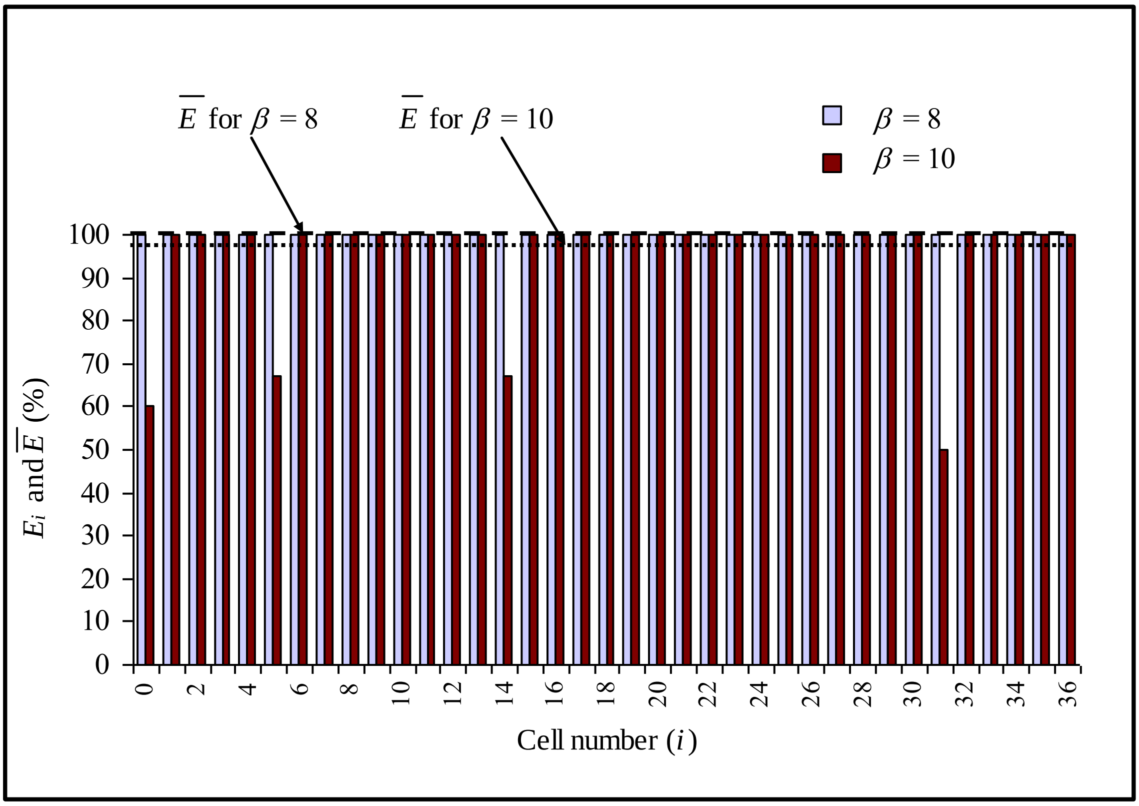


Figure (5.4). E_i versus cell number (i) for Scenario #1.

5.4.2 Scenario #2: Non-uniform traffic load

The second scenario represents a network with a nonuniform traffic load. It simulates a "hot-spot" in cell 8, in which the traffic load (□) is varied from 0.4 to 1.0 calls/sec in step of 0.2. Each cell around cell 8 (cells 1, 7, 9, 20, 21, and 22) has traffic load of 0.4 calls/sec. The traffic load of all other cells is 0.2 calls/sec. This scenario may represent a cellular network in an urban downtown area, where the traffic load is centralized in shopping or business districts. The results obtained for this scenario are presented in Tables (5.3) to (5.7) and Figures (5.5) to (5.7).

Tables (5.3) to (5.6) summarize the results for $C_{a,i}$, $C_{f,i}$, and E_i for values of □ as described above and uniform values of □□of 4, 6, 8, 10 channels/cell, respectively. In these tables, all suppressed cells have E_i equal to 100% for all values of □ and □□that are set for this scenario.

Table (5.3) shows the results for $C_{a,i}$, $C_{f,i}$, and E_i when □□□= 4 channels/cell and □□varies from 0.4 to 1.0 calls/sec for all cells ("hot-spot" cell and its first-tier neighbors). It shows that E_i is equal to 100% at all cells for □ of 0.4, 0.6, and 0.8 calls/sec. But, when □ becomes very high as it reaches 1.0 calls/sec, then E_i drops to 50% and less

, except for one cell (cell 22) where once again E_i and $\bar{E}$ of 100% are achieved. $\bar{E}$ for this simulation drops to 90% when it is taken over all cells within the network (i.e., for the 37 cells in Figure (3.1)), which is still acceptable for such high traffic load.

However, high efficiencies are achieved because the initial load distribution is relatively low and a lot of channels are available and can be allocated for the different cells. But, when the traffic load increases, not all the available channels can be allocated due to the constraints imposed on the system to avoid co-channel and adjacent channel interferences.

As shown in Tables (5.4) to (5.6) and summarized in Table (5.7), when □ increases to 6, 8, and 10 channels/cell, and □ increases from 0.4 to 1.0 calls/sec for cell 8 and all other cells are as specified in this scenario, more calls are dropped (i.e., channels can not be allocated) at the hot-spot cell and its firs-tier neighbors, and $\bar{E}$ is dropped down to its minimum value of 86% for □ = 10 channels/cell and □ = 1.0 calls/sec.

To avoid all forms of interferences, limited channels are allocated for the demanding cells when the traffic load is very high. However, if we can instantly re-allocate the channels, higher efficiencies would be achieved. But, this is left to future investigations. The results for Scenario #2 are also shown in Figures (5.5) to (5.7).

Table (5.3) - Results for Scenario #2.
Values of $C_{a,i}$, $C_{f,i}$, and E_i for a nonuniform traffic load and a uniform initial load (□□= 4).

Cell	□□= 0.4			□□= 0.6			□□= 0.8			□□= 1.0		
	$C_{a,i}$	$C_{f,i}$	E_i (%)	$C_{a,i}$	$C_{f,i}$	E_i (%)	$C_{a,i}$	$C_{f,i}$	E_i	$C_{a,i}$	$C_{f,i}$	E_i (%)
1	8	0	100	5	0	100	9	0	100	8	9	47
7	8	0	100	5	0	100	7	0	100	6	8	42
8	8	0	100	12	0	100	18	0	100	18	51	26
9	9	0	100	8	0	100	8	0	100	5	26	16
20	8	0	100	5	0	100	7	0	100	6	6	50
21	8	0	100	8	0	100	7	0	100	6	7	46
22	8	0	100	9	0	100	8	0	100	7	0	100

Table (5.4) - Results for Scenario #2.

Values of $C_{a,i}$, $C_{f,i}$, and E_i for a nonuniform traffic load and a uniform initial load (□□= 6).

Cell	□□= 0.4			□□= 0.6			□□= 0.8			□□= 1.0		
	$C_{a,i}$	$C_{f,i}$	E_i (%)	$C_{a,i}$	$C_{f,i}$	E_i (%)	$C_{a,i}$	$C_{f,i}$	E_i	$C_{a,i}$	$C_{f,i}$	E_i (%)
1	6	0	100	7	0	100	6	0	100	4	10	29
7	7	0	100	6	0	100	6	0	100	8	3	73
8	8	0	100	10	0	100	14	8	64	16	44	27
9	7	0	100	7	0	100	5	0	100	5	3	63
20	6	0	100	5	0	100	7	0	100	6	6	50
21	9	0	100	7	0	100	7	0	100	8	0	100
22	7	0	100	7	0	100	8	0	100	6	12	33

Table (5.5) - Results for Scenario #2.

Values of $C_{a,i}$, $C_{f,i}$, and E_i for a nonuniform traffic load and a uniform initial load (□□= 8).

Cell	□□= 0.4			□□= 0.6			□□= 0.8			□□= 1.0		
	$C_{a,i}$	$C_{f,i}$	E_i (%)	$C_{a,i}$	$C_{f,i}$	E_i (%)	$C_{a,i}$	$C_{f,i}$	E_i	$C_{a,i}$	$C_{f,i}$	E_i (%)
1	5	0	100	6	7	46	6	0	100	3	8	27
7	5	0	100	7	0	100	6	5	55	4	0	100
8	6	0	100	8	4	67	10	73	12	9	175	5
9	6	0	100	5	6	46	4	0	100	3	9	25
20	5	0	100	5	6	46	4	5	44	4	2	67
21	6	0	100	5	0	100	6	0	100	5	0	100
22	7	0	100	5	4	56	4	1	80	6	4	60

Table (5.6) - Results for Scenario #2.

Values of $C_{a,i}$, $C_{f,i}$, and E_i for a nonuniform traffic load and a uniform initial load

(□□= 10).

Cell	□□= 0.4			□□= 0.6			□□= 0.8			□□= 1.0		
	$C_{a,i}$	$C_{f,i}$	E_i (%)	$C_{a,i}$	$C_{f,i}$	E_i (%)	$C_{a,i}$	$C_{f,i}$	E_i	$C_{a,i}$	$C_{f,i}$	E_i (%)
1	4	12	25	5	4	56	5	6	46	3	4	43
5	3	0	100	3	0	100	2	0	100	1	1	50
7	2	11	15	3	8	27	4	0	100	2	6	25
8	2	10	17	5	64	7	7	162	4	7	241	3
9	4	8	33	5	0	100	4	7	36	2	10	17
15	2	0	100	2	0	100	2	0	100	1	1	50
20	6	10	38	6	3	67	4	0	100	4	11	27
21	5	0	100	5	0	100	3	0	100	2	0	100
22	1	0	100	4	0	100	3	0	100	2	1	67

Table (5.7) - Results for Scenario #2.

Values of $\overline{C_a}$, $\overline{C_f}$, $\overline{E}$, and their respective standard deviations (□$_a$, □$_f$, and □$_e$) for nonuniform traffic loads and various uniform initial loads (□ = 4, 6, 8, and 10 channels/cell□□

□	$\overline{C_a}$	□$_a$	$\overline{C_f}$	□$_f$	$\overline{E}$ (%)	□$_e$
□□= 0.4						
4	4.432	1.951	0.000	0000	100	0.000
6	3.351	2.031	0.000	0.000	100	0.000
8	3.162	1.385	0.000	0.000	100	0.000
10	2.378	0.953	1.378	3.570	90	0.260
□□= 0.6						

4	4.297	1.956	0.000	0.000	100	0.000
6	3.703	1.913	0.000	0.000	100	0.000
8	2.595	1.818	0.730	1.924	94	0.170
10	2.757	1.234	2.135	10.562	93	0.207
□□= 0.8						
4	4.892	2.875	0.000	0.000	100	0.000
6	3.460	2.422	0.216	1.315	99	0.060
8	2.622	1.949	2.270	12.006	94	0.183
10	2.514	1.096	4.730	26.615	94	0.204
□□= 1.0						
4	4.432	2.588	2.892	9.425	90	0.237
6	5.000	2.186	2.108	7.582	91	0.215
8	2.757	1.403	5.351	28.738	92	0.236
10	1.865	1.159	7.432	39.551	86	0.286

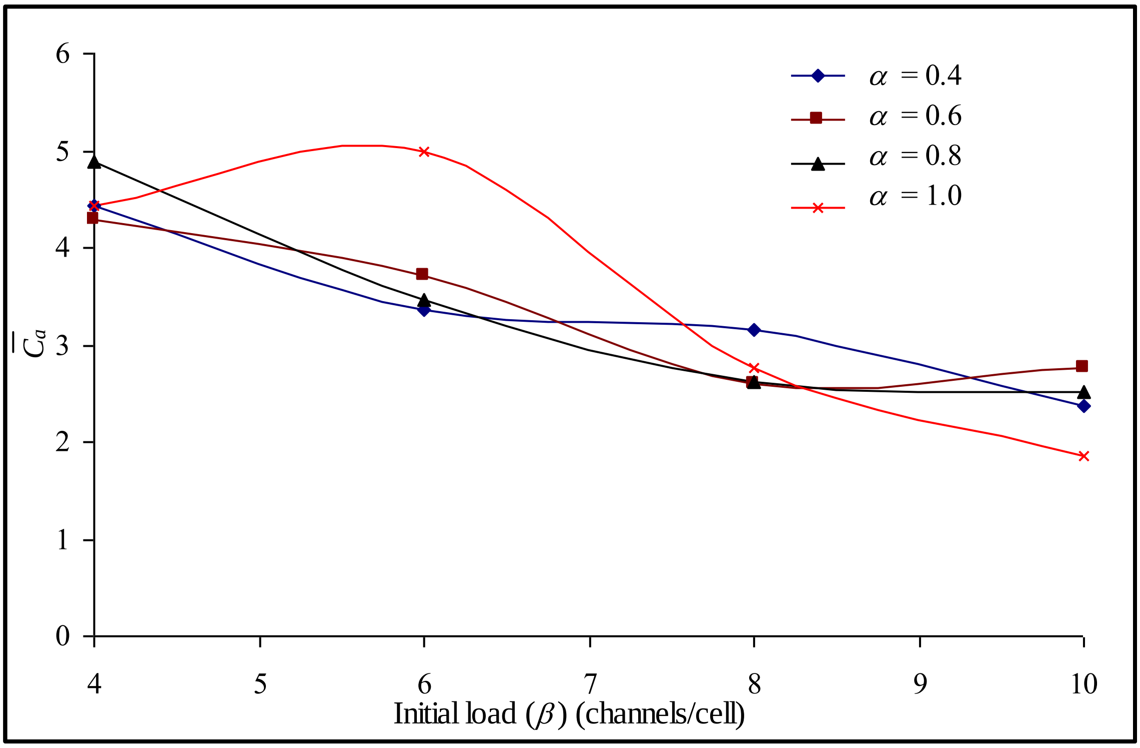


Figure (5.5). $\overline{C_a}$ versus □ for Scenario #2.

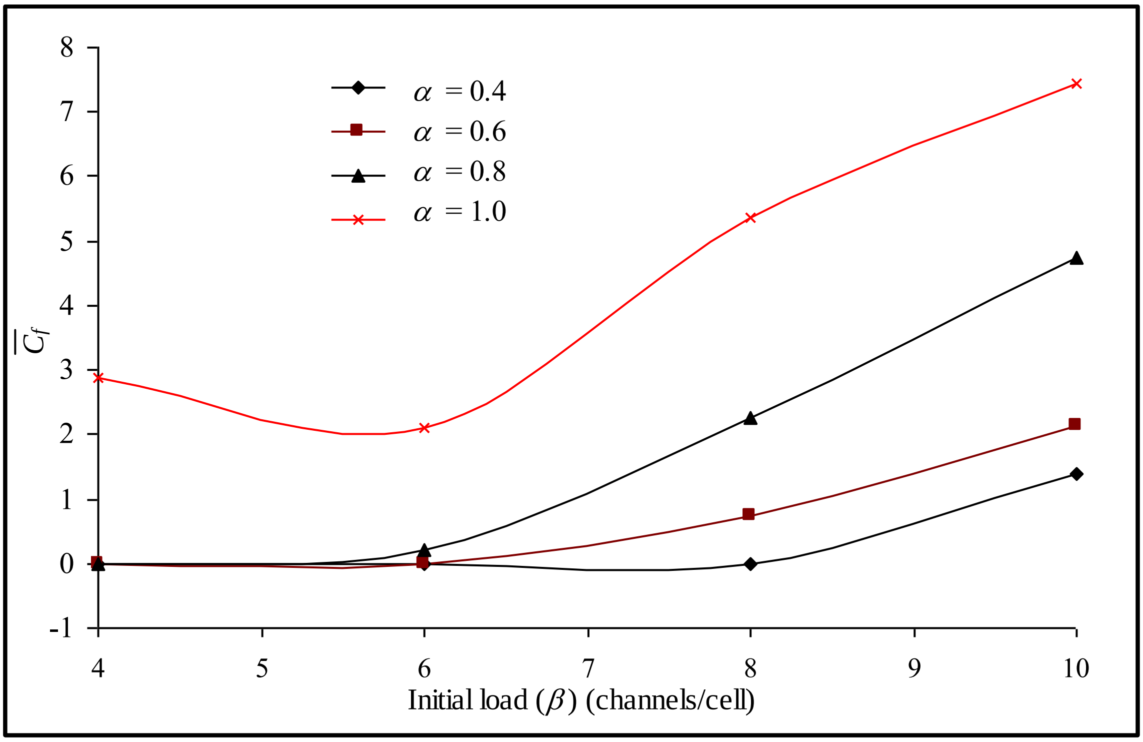


Figure (5.6). $\overline{C_f}$ versus □ for Scenario #2.

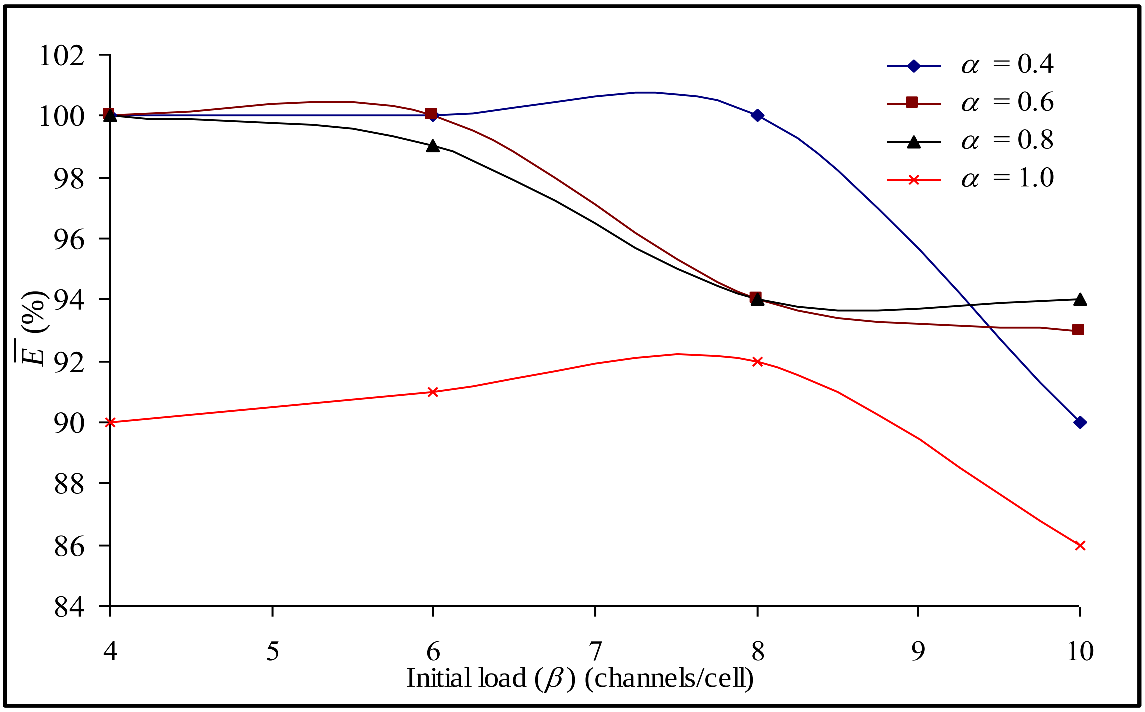


Figure (5.7). $\overline{E}$ versus □ for Scenario #2.

Finally, Table (5.8), summarizes the values of E_i for the hot-spot cell (cell number 8). The results clearly show that the efficiency reaches a very low value when □ = 1.0 calls/sec (high traffic load or rate of incoming calls) and □ = 10 channels/cell. This is due to the high traffic load, limited channels available, constraints imposed by the network to avoid all forms of interferences, and also the constraints imposed on the algorithm for not changing any of the allocated channels. The algorithm only manipulates the available channels in the best way to achieve the optimum performance.

Table (5.8) – Results for Scenario #2.
Values of E_i for cell number 8 (hot-spot cell).

□ (calls/sec)	□ (channels/cell)			
	4	6	8	10
0.4	100	100	100	17
0.6	100	100	67	7
0.8	100	64	12	4
1.0	26	27	5	3

Chapter 6

Conclusions and Recommendations for Future Work

6.1 Conclusions

The main conclusions of this thesis are:

1. An efficient and a reliable scheme for dynamic channel allocation (DCA) in limited bandwidth cellular communication networks (CCNs), such as the Global System for Mobile communication (GSM), was developed. The scheme is referred to as the DCA scheme. It utilizes a well-known distributed artificial intelligence (DAI) algorithm, namely, the asynchronous weak-commitment (AWC) algorithm. The AWC algorithm establishes a complete solution by extensive communication among a pattern of cells, where each cell in the pattern uses a unique set of channels (frequencies).

2. The performance of the new DCA scheme was evaluated in terms of a number of computed parameters, such as the average number of channels successfully allocated, the average number of channels that failed to be allocated, and the average allocation efficiency.

3. The DCA scheme presented an excellent performance to allocate and reuse channels efficiently under different network operation environments without violating the co-channel and adjacent channel interference constraints. This is mainly because the AWC algorithm could satisfy cells needs by intelligently allocating and reusing channels.

4. Two different scenarios representing different network operation environment were simulated, these are:

   a. Scenario #1. Uniform initial and traffic loads.

   b. Scenario #2. Uniform initial load and nonuniform traffic load.

Their results can be summarized as follows:

   a. The simulation results for Scenario #1 showed that an average allocation efficiency of 100% was achieved when the initial and the traffic loads were low and uniform (e.g., 8 channels/cell and 0.2 calls/sec, respectively) as in suburban or rural areas in GSM CCNs. The efficiency was

   b. reduced to 96% when the initial load was increased by 25% from 8 channels/cell to 10 channels/cell, and the traffic load remained unchanged at 0.2 calls/sec.

   c. The DCA scheme presented an excellent performance even at a heavily loaded network operation environment that simulates a GSM CCN in an urban downtown area, where the traffic load is centralized as in shopping or business districts. This was simulated in Scenario #2. The minimum average allocation efficiency achieved was 86% when the initial load at the hot-spot cell reaches 10 channels/cell and the traffic load is 1.0 calls/sec.

5. The performance of the DCA scheme is highly affected by the number of unused (free) channels remained after initialization, and the performance decreases as the number of free channels after initialization is decreased.

6. The DAI field could be used to utilize the limited resources available for CCNs efficiently, by making channel allocation process dynamic and intelligent.

7. The DCA scheme that is based on the AWC algorithm outperformed conventional DCA techniques; this fact came from the intelligent behavior of the AWC algorithm. It allocates channels in a dynamic and intelligent way so that it increases channels reusability and decreases conflict and interference with other cells, while conventional DCA techniques allocate channels randomly.

8. The results obtained for the *N*-Queen puzzle simulation showed that the AWC algorithm outperformed the ABT algorithm for all values of *N*. The AWC was faster as it needed relatively small number of cycles to find a solution when compared with ABT, also AWC solved the *N*-Queen puzzle, where *N* varying from 25 to 125 with success ratio equaled 100%, while ABT was unable to solve the puzzle when *N* increased in a reasonable time (number of cycles needed was greater than a certain maximum value), in addition the success ratio was decreasing as *N* increasing. It reached 8% at *N* equal to 125.

9. The introduction of the token-based mechanism enhanced the performance of the DCA scheme as it minimized

communication overheads among collaborative cells by passing channels that were currently allocated for other cells, and only allowed a cell holding the token to update its resources.

10. The delay, which is defined as the time taken by the token to pass through all cells within the pattern, was minimized by setting a certain threshold value for the additional channels or slots that each cell should keep for incoming calls. As a cell straightforwardly bypasses the token if it has enough resource for incoming calls.

6.2 Recommendations for Future Work

We suggest the following recommendations for future works,

1. To make the DCA scheme more efficient for allocating channels optimally in cases where there are many cells asking for more channels (high traffic load) and they can not find satisfactory channels, due to interference constraints (co-channel and adjacent channel interference), while there are unused channels in other cells, the DCA scheme could solve this situation by redistributing both the allocated and unused

2. channels in a way that satisfies all cells needs in the system. This seems impractical since some of the already allocated channels maybe in use (ongoing calls).

   This situation could be configured by redistributing the allocated channels in cells where the number of calls are small, or we could make the redistribution process dynamic through monitoring the traffic on all cells in the system. If dropping a specified number of calls for reallocating and redistributing channels will guarantee no more call dropping in a specified period of high traffic load time, then it may worth this cost. This needs to be studied carefully to know how much it will affect the efficiency of the DCA scheme in allocating channels and to figure out if this could be done without losing QoS.

3. In this thesis the hand-over process was not considered, we suggest implying the hand–over process in the DCA scheme, we expect this will improve the performance of the AWC algorithm in allocating channels.

4. One of the main strength keys of the AWC algorithm is the min-conflict heuristic function, we think that this heuristic could be

5. improved to increase the performance of the algorithm, or it could be aggregated with other heuristic functions to be more efficient in allocating channels.

References

[Abe 05] Gayan Abeysundara, “Dynamic Cellular Channel Allocation Using Intelligent Agents”, 100267102, COMP 4905 Honours Project Advisor: Dr. Tony White, School of Computer Science, April 2005.

[Agh 00] Khaldoun Al-Agha, "Resource Management in Wireless Networks Using Intelligent Agents", International Journal of Network Management Vol. 10, No. 1, pp. 29-39, 2000.

[Ahm 06] Rana Ahmad, "A Hybrid Channel Allocation Algorithm Using Hot-Spot Notification For Wireless Cellular Networks ", IEEE CCECE/CCGEI, pp. 891 – 894, 2006.

[An 07] J An, E L Hines, M S Leeson, L Sun, WRen and D D Iliescu, "Genetic Algorithms and Fuzzy Logic For Dynamic Channel Allocation in Cellular Radio Networks", IEEE Radio and Wireless Symposium, pp. 19 – 22, 2007.

[Bab 04] Rajeev Babbar, Abraham Fapojuwo, and Behrouz H. Far, "Agent-Based Resource Management in Hybrid Wireless Networks", IEEE, Vol. 3, pp. 1297–1300, 2004

[Bou 00] S. Boumerdassi, "An Efficient Reservation-Based Dynamic Channel Assignment Strategy", IEEE 3G Mobile Communication Technologies, 1st International Conference, No. 471, pp.352 – 355, 2000.

[Bej 04] Ramon Bejar, Carmel Domshlak, Cezar Fernandez, Carla Gomez, Bhaskar Krishnamachari, Bart Selman, Magda Valls, "Sensor Networks and Distributed CSP: Communication, Computation and Complexity", Artificial Intelligence Journal, Vol. 161, pp. 117–147, 2004.

[Bes 04] Christian Bessiere, Arnold Maestre, and partners, "Asynchronous Backtracking Without Adding Links: a new member in the ABT family", Artificial Intelligence Journal, Vol. 161, pp. 7–24, 2004.

[Cao 03] Guohong Cao, "Integrating Distributed Channel Allocation and Adaptive Handoff Management for QoS Sensitive Cellular Networks", Wireless Networks Journal, Vol. 9, No. 2, pp. 131-142, 2003.

[Car 96] C. Carciofi, G. Falciasecca, M. Frullone, N.P. Magnani, and V. Palestini, "Traffic-Adaptive and Interference-adaptive resource allocation schemes in GSM and DCS1800 Systems under realistic traffic and propagation conditions: a case study", IEEE Vehicular Technology Conference, Vol. 1, pp. 497-501, 1996.

[Das 06] Marcel Dasilva and Jose de Rezende, "A Dynamic Channel Allocation Mechanism for IEEE 802.11 Networks", IEEE International Telecommunications Symposium, pp. 225 – 230, 2006.

[Dut 08] Partha Dutta, Sharad Jaiswal, Debmalya Panigrahi, and Rajeev Rastogi, "A New Channel Assignment Mechanism for Rural Wireless Mesh Networks", IEEE 27th Conference on Computer Communications INFOCOM proceedings, pp. 2261 – 2269, 2008.

[Ema 06] Ibrahiem El Emary, "Managing a High Speed LAN Using Distributed Artificial Intelligence", Journal of Computer Science, Vol. 2, No. 1, pp. 92-96, 2006.

[Ezz 07] Redouane Ezzahir, Christian Bessiere, and partners, “Compilation Formulation for Asynchronous Backtracking with Complex Local Problems”, IEEE International Symposium on Computational Intelligence and Intelligent Informatics, pp. 205-211, 2007.

[Fal 04] Boi Faltings and Makoto Yokoo, "Introduction: Special Issue on Distributed Constraint Satisfaction", Artificial Intelligence, Vol. 161, No. 1-2, pp.1-5, 2004.

[Fel 06] Mark Felegyhazi, Mario Cagalj, Jean-Pierre Hubaux, "Multi-radio channel allocation in competitive wireless networks ", Proceedings of the 26th IEEE International Conference on Distributed Computing Systems Workshops, pp. 36 – 36, 2006.

[Fel 07] Mark Felegyhazi, Mario Cagalj, Shirin Saeedi Bidokhti, and Jean-Pierre Hubaux, "Non-cooperative Multi-radio Channel Allocation in Wireless Networks ", IEEE 26th International Conference on Computer Communication INFOCOM proceedings, pp. 1442 – 1450, 2007.

[Gar 06] Fernando P. Garcia, Angelo R. Brayner, Romildo J. Silva, Wandre Oliveira, and Ricardo Vasconcelos, "Indexing Mobile Objects: A Strategy for Supporting Dynamic Channel Allocation in Cellular Networks", IEEE International Conference on Wireless and Mobile Computing, pp. 284 – 291, 2006.

[Gol 07] S. Jamaloddin Golestani, Rajeev Rastogi, and Mark Smith, "Optimal Scheduling for Dynamic Channel Allocation in Wireless LANs ", IEEE 2nd International Conference on Communication Systems Software and Middleware, pp. 1-12, 2007.

[Gue 97] Lauro Ortigoza-Guerrero and Domingo Lara-Rodriguez, "CPMCB: A Suitable DCA Scheme For The Pan-European GSM System", IEEE 47th Vehicular Technology Conference, Vol. 3, pp. 1679 – 1683, 1997.

[Jen 93] Nick R. Jennings, "Commitments and Conventions: The Foundation of Coordination in Multi-Agent Systems", The knowledge Engineering Review, Vol. 8, No. 3, pp. 223-250, 1993.

[Jia 08] Peng Jiang, John Bigham, and Jiayi Wu, "Cooperative Geographic Load Balancing in Hybrid Cellular Networks", IEEE International Conference on Signal processing, Communications and Networking Madras Institute of Technology, pp. 288-294, 2008.

[Jie 06] Long-Mei Jie, Hong-Guang Sun, and partners, "Exploiting Csp Techniques In Temporal Graphplan", IEEE 5th International Conference on Machine Learning and Cybernetics, pp. 846-849, 2006.

[Jun 04] Hyuckchul Jung and Milind Tambe, "On Communication in Solving Distributed Constraint Satisfaction Problems", International Central and Eastern European Conference on Multi-Agent Systems, Vol. 3690, pp. 418–429, 2004.

[Kha 08] Lutfi Khanbary and Veo Vidyarthi, "A GA-Based Effective Fault-Tolerant Model for Channel Allocation in Mobile Computing", IEEE Transactions On Vehicular Technology, Vol. 57, No. 3, pp. 1823 - 1833, 2008.

[Khe 07] Mohammad Taghi Kheirabadi and Hossein Mohammadi, "MARP: A Multi-Agent Routing Protocol for Ad-hoc Network", IEEE Second International Conference on Communications and Networking in China, pp. 234 – 238, 2007.

[Kim 08] Dongkyu Kimt, Hano Wang, Jemin Lee, and Daesik Hongtt, "Analysis of Handoff Algorithm for Multihop Cellular Networks", IEEE 10th International Conference on Advanced Communication Technology, Vol. 1, pp. 29 – 32, 2008.

[Kog 07] Okuthe P. Kogeda and Johnson I. Agbinya, "Proactive Cellular Network Faults Prediction Through Mobile Intelligent Agent Technology", IEEE 2nd International Conference on Wireless Broadband and Ultra Wideband Communications, pp. 55-55, 2007.

[Kos 01] Zoran Kostic Ivana Maric, and Xiaodong Wang, “Fundamentals of Dynamic Frequency Hopping in Cellular Systems”, IEEE Journal on Selected Areas in Communications, Vol. 19, No. 11, pp. 2254-2266, 2001.

[Kos 02] Zoran Kostic and Nelson Sollenberger, “Performance and Implementation of Dynamic Frequency Hopping in Limited-Bandwidth Cellular Systems, IEEE Transactions on Wireless Communications, Vol. 1, No. 1, pp. 28-36, 2002.

[Kou 07] Iordanis Koutsopoulos, and Leandros Tassiulas, "Joint Optimal Access Point Selection and Channel Assignment in Wireless Networks", IEEE/ACM Transactions on Networking, Vol. 15, No. 3, pp. 521-532, 2007.

[Lee 07] Neung-Hyung Lee and Saewoong Bahk, "Dynamic channel allocation using the interference range in multi-cell downlink systems ", IEEE Wireless Communication and Networking Conference, pp. 1716 – 1721, 2007.

[Lei 06] D. J. Leith and P. Clifford, "Convergence of Distributed Learning Algorithms for Optimal Wireless Channel Allocation ", IEEE 45th Proceedings Conference on Decision & Control, pp. 2980 – 2985, 2006.

[Leu 07] Kin K. Leung, , Martin V. Clark, Bruce McNair, Zoran Kostic, Leonard J. Cimini, Jr., and Jack H. Winters, "Outdoor IEEE 802.11 Cellular Networks: Radio and MAC Design and Their Performance", IEEE Transactions On Vehicular Technology, Vol. 56, No. 5, pp. 512 - 516, 2007.

[Mah 07] Mehdi Mahdavi, Robert M. Edwards, and Charalampos V. Ladas, "On the Effect of a Random Access Protocol on the Performance of the Session-Based Data Subsystem in GSM/GPRS", IEEE Transactions On Vehicular Technology, Vol. 56, No. 4, pp. 1781 - 1796, 2007.

[Mar 07] Brian L. Mark and Alexe E. Leu, "Local Averaging for Fast Handoffs in Cellular Networks", IEEE Transactions on Wireless Communications, Vol. 6, No. 3, pp. 866-874, 2007.

[Mar 08] N. V. Marathe, U. B. Desai, and S. N. Merchant, "Base Station Selection Strategy in Multihop Cellular Networks: A New Approach", IEEE International Conference on Signal processing, Communications and Networking Madras Institute of Technology, pp. 401-404, 2008.

[Mod 01] Pragnesh Jay Modi, Hyuckchul Jung, Milind Tambe, Wei-Min Shen, and Shriniwas Kulkarni, "Dynamic Distributed Resource Allocation: Distributed Constraint Satisfaction Approach", Springer Berlin / Heidelberg, Vol. 2333/2002, pp. 264 - 276, 2001.

[Moh 07] A. Mohsenian-Rad, and Vincent Wong, "Joint Logical Topology Design, Interface Assignment, Channel Allocation, and Routing for Multi-Channel Wireless Mesh Networks ", IEEE Transactions On Wireless Communications, Vol. 6, No. 12, pp. 4432 – 4440, 2007.

[Nav 02] Navarro, J., Martinez, J., and partners, "Signaling capacity in GSM & (E)GPRS networks", IEEE 55th Vehicular Technology Conference, Vol. 4, pp. 1970– 1974, 2002.

[Per 07] Artur Pereira, Joaquim Bastos, Paulo Marques, and Atílio Gameiro, "Analysis of Communication Opportunities in UMTS Cellular Networks", Second IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks, pp. 99 – 102, 2007.

[Per 07] Artur Pereira and Atilio Gameiro, "Simulation-based Evaluation of Spectrum Opportunities in UMTS Cellular Networks", IEEE 16th Mobile and Wireless Communication Summit, pp. 1-5, 2007.

[Pic 07] Gauthier Picard, Marie-Pierre Gleizes, and partners, "Distributed Frequency Assignment Using Cooperative Self-Organization", IEEE 1st International Conference on Self-Adaptive and Self-Organizing Systems, pp. 183-192, 2007.

[Pri 97] Francesco Delli Priscoli, Nicola Pio Magnani, Valerio Palestini, and Fabrizio Sestini, "Application of Dynamic Channel Allocation Strategies to the GSM Cellular Network", IEEE Selected Areas In Communications, Vol. 15, No. 8, pp. 1558-1567, 1997.

[Rah 08] Keyvan RahimiZadeh, Mehdi Dehghan, Seyed Ali Hosseininezhad, and Abbas Dehghani, "A Cellular-Based Routing Algorithm for Ad-Hoc Wireless Networks", IEEE 10th International Conference on Advanced Communication Technology, Vol. 1, pp. 317 – 321, 2008.

[Ras 07] Sherif Rashad, Mehmed Kantardzic, and Anup Kumar, "MSP-CACRR: Multidimensional Sequential Patterns Based Call Admission Control and Resource Reservation for Next-Generation Wireless Cellular Networks", IEEE Symposium on Computational Intelligence and Data Mining, pp. 552 – 559, 2007.

[Rob 00] Robert, L., Pissinou, N., and partners, "Third generation wireless network: the integration of GSM and Mobile IP", IEEE Wireless Communications and Networking Conference, Vol. 3, pp. 1291 – 1296, 2000.

[Rus 03] Stuart Russell and Peter Norvig, "Artificial Intelligence A Modern Approach", Prentice Hall, Second Edition, ISBN-10: 0137903952, ISBN-13: 978-0137903955, 2003.

[Sae 05] Koragod Saenchai, Luigi Benedicenti, and partners, "A Dynamic Extension Of The Asynchronous Weak commitment Search Algorithm", IEEE Canadian Conference on Electrical and Computer Engineering, pp. 1112-1115, 2005.

[Sal 01] Matti Salmenkaita, Jose Gimenez and Pablo Tabia, "A Practical DCA Implementation for GSM Networks: Dynamic Frequency and Channel Assignment", IEEE Vehicular Technology Conference, Vol. 4, pp. 2529– 2533, 2001.

[Sar 00] Behcet Sarikaya, " Packet Mode in Wireless Networks: Overview of Transition to Third Generation", IEEE Communications Magazine, Vol. 38, No. 9, pp. 164-172, 2000.

[Sha 06] Nazaraf Shah1, Kuo-Ming Chao1, Nick Godwin1, Anne James1. C-F Tasi2, "An Empirical Evaluation of a Sentinel Based Approach to Exception Diagnosis in Multi-Agent Systems", IEEE Proceedings of the 20th International Conference on Advanced Information Networking and Applications, Vol. 1, pp. 6, 2006.

[Sil 04] Marius-Calin Silaghi and Boi Faltings, "Asynchronous Aggregation And Consistency In Distributed Constraint Satisfaction", Artificial Intelligence, Vol. 161, pp. 25–53, 2004.

[Sil 06] Marius-Calin Silaghi, “Framework for modeling reordering heuristics for asynchronous backtracking”, IEEE International Conference on Intelligent Agent Technology, pp. 529-536, 2006.

[Spy 07] Ioannis Spyropoulos and James R. Zeidler, "Dynamic Channel Allocation and Space-Time LMMSE Joint Detection in a TDD/CDMA Cellular Network with Traffic Asymmetry ", IEEE 65th Vehicular Technology Conference, pp. 3066 – 3070, 2007.

[Sta 05] W. Stallings, **Wireless Communications & Networks**, Prentice-Hall, 2th Edition, 2005.

[Sun 07] Jingyuan Sun, "Uplink Capacity Enhancement In Two-Hop Cellular Networks With Limited Mobile Relays", IEEE 15th Proceedings Workshop on Local and Metropolitan Area Networks, pp. 134 – 138, 2007.

[Tan 03] A. Tanenbaum, **Computer Networks**, Prentice-Hall, 4th Edition, 2003.

[Tok 06] Sanjiv Tokekar and Neetesh Purohit, "Analysis of a New Fixed Channel Allocation Scheme for a Sectorized GSM Cellular Network", IEEE International Conference on Wireless and Optical Communications Networks, pp. 5, 2006. .

[Tra 08] Nguyen H. Tran and Choong Seon Hong, "Joint Scheduling and Channel Allocation in Wireless Mesh Networks", IEEE Consumer Communications and Networking Conference proceedings, pp. 760 – 764, 2008.

[Ver 07] Roberto Verdone, and Alberto Zanella, "On the Effect of User Mobility in Mobile Radio Systems With Distributed DCA", IEEE Transactions On Vehicular Technology, Vol. 56, No. 2, pp. 874 - 887, 2007.

[Wal 04] Richard J. Wallace and Eugene C. Freuder, "Constraint-Based Reasoning And Privacy/Efficiency Tradeoffs In Multi-Agent Problem Solving", Artificial Intelligence, Vol. 161, pp. 209–227, 2004.

[Wan 06] Zhipeng Wang and P. Takis Mathiopoulos, "On the Performance Analysis of Dynamic Channel Allocation With FIFO Handover Queuing in LEO-MSS", IEEE Transactions On Communications, VOL. 53, NO. 9, pp. 1443 – 1446, 2005.

[Wan 07] Jinsu Wang, Sharad Mehrotra, and Nalini Venkatasubramanian, "PBCA - Prediction Based Channel Allocation", IEEE GLOBECOM proceedings, pp. 4801 – 4806, 2007.

[Wan 98] Wai-Leung Wan and Wing-Shing Wong, "A Heuristic Algorithm for Channel Allocation of Multi-rate Data in Hybrid TDMA/FDMA Digital Cellular Systems", IEEE 9th International Symposium on Personal Indoor and Mobile Radio Communications, Vol. 2, pp. 853-858, 1998.

[Wei 99] Gerhard Weiss, **Multiagent Systems: A Modern Approach to Distributed Artificial Intelligence**, MIT Press, ISBN 0-262-23203-0, 1999.

[Wil 01] Steven Willmott, "Coordination Structures for the Intelligent Control of Dynamic Distributed Systems", Thesis No.2478, Lausanne Switzerland EPFL, 2001.

[Wu 00] Xiaoxin Wu, Biswanath Mukherjee, and S. Gary Chan, "MACA-An Efficient Channel Allocation Scheme in Cellular Networks", IEEE Global Telecommunication Conference, Vol. 3, pp. 1385-1389, 2000.

[Yan 05] Jianchang Yang, Qiangfeng Jiang, D. Manivannan, and Mukesh Singhal, "A Fault-Tolerant Distributed Channel Allocation Scheme for Cellular Networks", IEEE Transactions On Computers, Vol. 54, No. 5, pp. 616-629, 2005.

[Yan 07] Jianchang Yang, Qiangfeng Jiang, and D. Manivannan, "A Fault-Tolerant Channel-Allocation Algorithm for Cellular Networks With Mobile Base Stations ", IEEE Transactions On Vehicular Technology, Vol. 56, No. 1, pp. 349 – 361, 2007.

[Yan 07] Jianchang Yang and D. Manivannan, "Performance Comparison of Two Channel Allocation Approaches: Channel Pre-allocation Vs. Non-Pre-allocation ", IEEE Proceedings of the Third International Conference on Wireless and Mobile Communications, pp. 61 – 61, 2007.

[Yok 00] Makoto Yokoo and Katsutoshi Hirayama, "Algorithms for Distributed Constraint Satisfaction: A Review", Autonomous Agents And Multi-Agent Systems, Vol.3, No.2, pp.198-212, 2000.

[Yok 92] Makoto Yokoo, E.H. Durfee, Toru Ishida, and Kazuhiro Kuwabar, "Distributed Constraint Satisfaction for Formalizing Distributed Problem Solving", IEEE 12th International Conference on Distributed Computing Systems, pp. 614–621, 1992.

[Yok 94] Makoto Yokoo, "Weak-Commitment Search for Solving Constraint Satisfaction Problems", 12th International Conference Artificial Intelligence, pp. 313–318, 1994.

[Yok 95] Makoto Yokoo, "Asynchronous Weak-Commitment Search for Solving Distributed Constraint Satisfaction Problems", Springer-Verlag 1st International Conference on Principles and Practice of Constraint Programming, pp. 88–102, 1995.

[Yok 98] Makoto Yokoo, Edmund H. Durfee, Toru Ishida, and Kazuhiro Kuwabar, "The Distributed Constraint Satisfaction Problem: Formalization and Algorithms", IEEE Transactions On Knowledge And Data Engineering, Vol. 10, No. 5, pp. 673-685, 1998.

[Yu 97] Oliver T. W. Yu and Victor C. M. Leung, "Adaptive Resource Allocation for Prioritized Call Admission over an ATM-Based Wireless PCN", IEEE Journal On Selected Areas In Communications, Vol. 15, No. 7, pp. 1208-1225, 1997.

[Yub 00] ShaoYubin, HuangHuan, Long Hua, and Yao Shaowen, "Performance Analysis Of The Minimum Call Blocking Probability For Dynamic Channel Allocation In Mobile Cellular Networks", IEEE International Conference on Communication Technology Proceedings, Vol. 1, pp. 269-273, 2000.

[Zha 05] Weixiong Zhang, Guandong Wang, Zhao Xing and Lars Wittenburg, “Distributed Stochastic Search and Distributed Breakout: Properties, Comparison and Applications to Constraint Optimization Problems in Sensor Networks”, Journal of Artificial Intelligent, Vol. 161, Issues 1-2, pp. 55-87, 2005.

[Zha 89] Zhang, M. and Yum, TS. P, "Comparisons of Channel Assignment Strategies in Cellular Mobile Telephone Systems", IEEE Transactions On Vehicular Technology, Vol. 38, No. 4, pp. 211-215, 1989.

[Zha 99] Yongbing Zhang, "An Efficient Channel Allocation Strategy For Mobile Cellular Networks", IEEE 49th Conference on Vehicular Technology, Vol. 1, pp. 611-616, 1999.

[Web 1] http://www.gsmworld.com/documents/prd/in0510.pdf, Available February 2008.

[Web 2] http://www.gsmworld.com/documents/prd/ir2142.pdf, Available February 2008.

[Web 3] http://www.gsmworld.com/documents/nfc/gsma_nfc2_wp.pdf , Available February 2008.

[Web 4] http://www.gsmworld.com/documents/health/interference.pdf, Available February 2008.

[Web 5] http://www.gsmworld.com/documents/wireless_evolution_insider/wei_jan04.pdf, Available February 2008.

[Web 6] http://www.ericsson.com/hr/services/ GSM System Survey, Available February 2008.