PARTICLE SWARMS AND THE FREQUENCY ASSIGNMENT PROBLEM

by

WILLIAM BEZUIDENHOUT

DISSERTATION

submitted in the fulfilment
of the requirements for the degree

MAGISTER SCIENTIAE

in

INFORMATION TECHNOLOGY

in the

FACULTY OF SCIENCE

at the

UNIVERSITY OF JOHANNESBURG

SUPERVISOR: DR. G.B. O'REILLY

JUNE 2010

Contents

Part I

Background

Chapter 1

Introduction

1.1 Test

blah blah blah blah

Chapter 2

Cellular Technology

2.1 Introduction

In the information age we currently live in almost every device has some sort of wireless technology it uses to provide a specific service. Radios for audio entertainment; Television remotes to change channels; Cellular phones for communication; Wireless access points to create wireless LAN's [2]. Wireless technology is now part of our everyday life.

The popularity and rapid adoption of wireless technology hasn't been kind to the management, planning and operation of wireless networks; It has actually worsen a problem known as the Frequency Assignment problem (FAP) which is present in all forms of wireless communication especially in GSM Cellular networks.

In this chapter we will start of by giving a brief history of the Frequency Assignment Problem. After the brief history overview we will give a description of the problem and explain some of the underlying concepts needed to understand the Frequency Assignment Problem. We will then move on to discuss the different variants present in the current domain. Further more we will then discuss the two underlying approaches used in solving the Frequency Assignment Problem (FAP).

2.2 GSM Networks

The General System for Mobile Communications (GSM) is a system for multi-service cellular communication which is capable of providing voice as well as data services. Most cellular networks in operation are GSM based. The primary service that GSM caters for is voice communication, but other data services such as Short Message Service (SMS), Multimedia Message Service (MMS) and Internet connectivity services such as GPRS are becoming more important [15].

GSM is one of the most widely used radio communications technologies, which is why we need to look at the history behind it in order for us to understand the domain of radio communication better.We will now present a brief history of the GSM network specification.

2.2.1 A Brief History of GSM Networks

In the early 1981 a group known as the Groupe Speciale Mobile (GSM) was established to develop a Europe wide radio communication system using the reserved 900 MHz band[1].

At the start of the GSM specification in the early 1980's it was initially thought that the system would be analogue based, but this soon changed with the *Integrated Service Digital Network* (ISDN) specification nearing completetion. As such the GSM specification started following much of the same design principles and access protocols that ISDN exhibited. After the completion of the ISDN specification and the advantages it brought to the field, it became unofficially clear that GSM would be based on digital transmission and that speech would be represented by a digital stream of 16 kbits/s [44].

Before the switch to digital transmission was finalized the GSM first wanted to evaluate the spectral efficiency of analogue and digital based transmission. Spectral efficiency plays an important part in wireless communication since the radio spectrum is a limited resource and whichever transmission technology is used, should maximise the utilization of the spectrum. Maximum utilisation is an important problem which we will discuss in detail in later sections of this chapter. The Spectral evaluation was conducted over a period of 3 years from 1984-1987. In 1987 a report was published about the 3 year evaluation and subsequently it became official that the

[1]In 1990 the United Kingdom requested that 1800MHz band be added to the scope of the GSM standard group. This variant of the GSM specification became known as the *Digital Cellular System 1800* (DCS1800) [44].

GSM system would be digital based using *Time Division Multiple Access* (TDMA) [40, 44].

By the early 1990s GSM became an evolving standard and the first GSM based network was demonstrated in 1991[2]. The following year a number of GSM networks were operating in Europe due to mobile terminals / equipment capable of operating on the networks becoming more widely available to the general public. In the same year an operator in Australia became the first non-European operator to implement a GSM based network [15].

The collective subscriber base of GSM networks surpassed the million subscriber mark in 1993. Due to this phenomenal growth in GSM network use, numerous extensions were made to the GSM specification. Some of the extensions that were made are the following [15]:

- Half rate speech telephony
- Improved SMS
- Line Identification
- Call waiting
- Call holding

The specification with these extensions defined is known as the GSM Phase 2. As the world shifted towards more digital and data intensive services it became difficult to deliver these services over GSM networks. This difficulty was due to the restriction that data could only be transmitted at 9.6 Kbps. A move to eliminate this restriction was made with the specification of GSM Phase 2+.

The new specification defined new technologies such as General Packet Radio System (GPRS) and *Enchanced Data rate for GSM Evolution* (EDGE) which were designed with the primary goal of making more bandwidth available for data transmission. These new technologies have an inherent requirement that there be a higher signal to noise ratio present at transceivers. This requirement has an impact that effects radio interfaces and more importantly Frequency planning [15].

[2] Near the end of 1991 the GSM group was renamed to *Speciale Mobile Group* (SMG) to eliminate confusion with the standard and the group.

The actual signal to noise ratio at a receiver is dependent on a number of factors that include [2]:

- Frequency used at the transceiver
- Strength of the signal
- Weather conditions
- Shape of the surrounding environment
- Direction of the transmission

Even taking these factors into account the calculation of the signal to noise ratio at a transceiver is not trivial. For a more in depth discussion on the calculation the reader is directed at the survey by [2].

As the GSM standard matured as a cellular technology, industry experts already began specification of the next generation of cellular networks which would in time, replace the GSM cellular system. The specification of a new standard is considered to be a natural evolution of the technology. Each standard is designed with specific use cases in mind as to what its users might want to do as well as what is possible with the technology at the designers disposal. As time goes by, the technology improves and users habits and needs change, thus the standard must be improved upon to serve these new needs and incorporate new technology.

The *Universal Mobile Telecommunications System* (UMTS) can be considered the 3rd generation (3G) of cellular networks. UMTS was designed from the beginning to operate in parallel with the legacy GSM system. This decision was made to make the deployment of the system as hassle free as possible for the network operators. The first standard of the UMTS was issued in the beginning of 2000 and subsequently most modern networks are based on it or are migrating their networks to it.

UMTS is a large improvement of the GSM in two areas namely Data Transmission bandwidth and Frequency Planning due to UMTS utilising *DS-CDMA* (direct sequence code division multiple access) and *WCDMA* (Wide Band Code Division Multiple Access). The higher data transmission speed (2 Mb/s) can be attributed to UMTS using the DS-CDMA scheme. The scheme also allows more users to be served than previous generation of networks. A direct consequence of WCDMA which sends information over a wide-band of 5 MHz is that no frequency planning problem comparable to that of GSM has to be solved [15, 69].

In this section we gave a brief overview of the history of the GSM Network specification. In the next section we will present an explanation of the topology of GSM network as well as look at the underlying problems present in a GSM networks.

2.3 Topology of a GSM Network

GSM networks consists of a variety of different subsystems to realise the goal of establishing a radio communication link between two parties. The hierarchy of systems and their respective connections to each other is illustrated in figure 1. We will now briefly explain each subsystem.

Mobile Station (MS)

A Mobile station (MS) as it is defined in the GSM spec refers to any mobile device that is capable of of making and receiving calls on a GSM network. The MS is the main gateway for a user to gain access to the GSM network. The MS has two features which play an important role throughout the GSM Network, namely:

Subscriber Identification Module (SIM) — Usually inserted into a mobile devices. The SIM contains the *International Mobile Subscriber Identity* (IMSI) and is used throughout the network for Authentication as well as being a key part in providing encrypted transmissions.

International Mobile Equipment Identity(IMEI) — Used to identify mobile station equipment. Primarily used in the denial of service to equipment that has been blacklisted[3] and tries to gain access to the network.

The MS has the capability to change the transmission power is uses from its base value to a maximum value of 20 mW. The change in transmission power is automatically set to the lowest level by the Base Transceiver Station to ensure reliable communication after evaluating the signal strength as measured by the MS [40].

[3]Equipment can be blacklisted for a variety of reasons e.g. theft

2.3.1 Base Station Subsystem (BSS)

According to the GSM Phase 2+ specification this system is viewed by the *Mobile Switching Centre* (MSC) through an Abis radio interface as the system responsible for communicating with Mobile Stations in a particular location area. The BSS usually consists of one *Base Station Controller* (BSC) with one or more *Base Transceiver Stations* (BTS) which it controls. The communication link between the MSC and BSC is the called the A-interface and the communication link between the BSC and BTS is called the Abis interface. The definition of these communication interfaces is beyond the scope of this disseration, the interested reader is directed to the book *GSM System Engineering* by Asha Mehrotra. A BTS has similar equipment to that of a Mobile Station. Both have transceivers, antennas and the necessary functions to perform radio communication.

In a GSM network the Service Area (SA) is subdivided into Location Areas(LA's) which are then futher divided into smaller radio zones called cells [56]. A cell is served by only one BTS and is usually regarded to be in the center of a cell. Even though cells are modelled as being hexagons (See figure 3) the actual coverage area of a cell has no predefined regular shape. Futhermore a cell is divided into 1 to 3 service sectors and each sector is allocated an antenna/transceiver [40]. Depending on how many sectors are at a cell, the operating angle of the antennae needs to be adjusted accordingly to ensure 360 degree service. If there is only one sector an omni-directional antenna is used, otherwise the antennae operating angle are adjusted to $\frac{360^\circ}{n}$ where n is the amount of antennae [15].

Each sector operates one or more elementary transceivers called TRXs. The amount of TRXs per sector is determined by the expected peak traffic demand that the cell must be able to handle. Each TRX can handle 7 to 8 communication links or calls in parallel except the first TRX, which handles fewer calls than normal due to it being responsible for transmitting cell organisation and protocol information [15]. TRXs are able to handle 7-8 calls in parallel due to the use of *Frequency Division Multiplexing* (FDM) and *Time Division Multiplexing* (TDM) schemes. TRXs are assigned channels which enable them to provide conversion between digital traffic data on the network side and the radio communication between Mobile Stations and the GSM network. [8, 36]

2.3.2 Mobile Switching Centre (MSC)

The MSC is at the heart of cellular switching system and forms part of the *Network Switching Subsystem* (NSS). The MSC is responsible for the setting up, routing and supervision of calls between GSM subscribers. The MSC has interfaces on the one side to communicate with GSM subscribers (through the BSS) and on the other it has interfaces to communicate with external networks. The MSC interfaces with external networks to utilise their superior capability in data transmission as well as for the signalling and communication with other GSM entities [44].

The most basic functions that an MSC is responsibile for in a network are the following [49]:

- Voice call initialization,routing,control and supervision between subscribers.
- Handover process between two cells.
- Location updating.
- MS authentication.
- SMS delivery.
- Charging and Accounting of services used by subscriber.
- Notification of other network elements.
- Administration input or ouput processing functions.

To achieve most of these functions the MSC has an integrated *Visitor Location Register (VLR)* database that stores call setup information for any MS that is currently registered for service with the MSC [44, 49].

The VLR retrieves this information from the *Home Location Register (HLR)* which contains all the registered GSM subcriber information for the network. This information enables the MSC to quickly retrieve the nesaccery information to setup a call between two entites [40, 56].

A requirement for being able to communicate with other network elements such as *Public Switching Telephone Networks* (PSTN) is the ability to multiplex and demultiplex signals to and from such network elements.

This operation is a necacity, since the incoming or outgoing connection bitrate from the source entity might either be to low or to high for the receiving entity.

A typical scenario where this operation proves vital is when a mobile subcriber makes a call to a subcriber on a PSTN. The connection bit rate needs to be changed at the MSC from a wireless connection bitrate to a bitrate suitable for transmission over a PSTN.

2.3.3 Network Databases

The HLR, AUC (Authentication Center) and EIR (Equipment Identity Register) are the 3 'back-end' databases which stores and provides information for the rest of the GSM Network. We will now briefly discuss what the purpose of each database is and its core functions.

Home Location Register (HLR) The HLR is a database that permenantly stores information pertaining to a given set of subscribers. The HLR needs to store a wide range of subcriber parameters because it is the reference source for anything GSM subscriber related in the network.

Subscriber parameters that are stored in the database include: Billing information, routing information, identification numbers, authentication parameters,subscribed services. The following information is also stored but the information is of a temporary nature and can change at anytime: Current VLR and MSC the subscriber is registered with; Wheter the subscriber is roaming [40].

Authentication Center (AUC) The Authentication center is the entity in the GSM network that performs security functions and thus stores information that enables it to provide secure over the air communication. The information that is stored contains authentication information as well as keys that are used in ciphering of information.

During an authentication procedure no ciphering key is ever transmitted over the air, instead a challenge is issued to the mobile who needs to be authenticated. This callenge requires the mobile station to provide the correct *Signed Response*(SRES) with regard to the random number generated by the AUC. The random number and ciphering keys that are used change with each call that is made, thus an attacker would gain nothing by intercepting a key, since it will change with the next call [40].

Each mobile that is registered in the HLR database needs to be authenticated and each call that is instansiated needs to retrieve keys from the AUC to establish a secure communication link. The AUC is sometimes included with HLR to allow for fast communication between the two entities [40].

Equipment Identity Register (EIR) The EIR is a database that stores the IMEI numbers of all registered mobile equipment that has accessed the network. Only information about the mobile equipment is stored, nothing about the subscriber or call is stored in the database.

Typically there is only one EIR database per network and interfaces to the various HLR database contained in the network. The IMEI's are grouped into 3 categories: *White List*, *Black List* and the *Gray List*. The White list contains only the IMEI numbers of valid MS's; the Black List stores the IMEI numbers of equipment that has been reported stolen and the Gray List stores the IMEI numbers of equipment that has some fault (faulty software, wrong make of equipment).

2.3.4 GSM Network Management entities

In a GSM network most of the elements that form part of and make the network function are often distributed in a wide geographical area to provide the best network converage for the customer.

For a network to function properly and efficiently network engineers need to be kept up to date on the state of the network and be alerted if *any* problems occur. For this purpose there exists two systems in the GSM network architecture that allows for this functionality required by network engineers.

The one system is called the Operations and Management center which is responsible for centralized regional and lcoal operational and maintenance activities. The other system called the Network Management System unlike the OMC provides global and centralized managemenent for operations and maintenance of the network suppored by the OMCs [40].

We will not discuss the OMC and NMC in a bit more detail where we'll define the most critical functions they peform.

Operational and Managemenet Center The OMC is capable of communicating with GSM entities using two protocols namely SS7 and X.25. The SS7 protocol is usually used when the OMC is communicating within the GSM network over short and medium distances. The X.25 protocol is used for large external data transfers. All communication where the OMC is involved typically occurs over fixed line networks and/or leased lines. The OMC is usually used for day to day operation of a network [40].

The OMC has support for alarm handling. An alarm in a GSM network goes of whenever a predefined expected condition does occurs. Engineers are able to define the severity of an alarm which defines who and what is futher alerted when and if the alarm is escalated to a higher level [40].

To give one an idea of when and why and alarm goes of consider the following scenario: The MSC controls a set of BSS systems. Now for some reason a certain region experiences a power blackout. All the BSS affected by the blackout switch over to reserve power if availble. A first alarm is sounded to let the engineers / network know that the BSS is using reserve power. When the BSS reserve power is depleted the MSC sounds an alarm letting the network know that a specific BSS cannot be contacted.

Typically in the above scenario the severity of the first alarm will be of a medium priority. The second alarm is much more serious and its severity will be of a high priority.

The OMC is also capable of fault management in the GSM network. The OMC is able to activate,deactivate,remove and restore a service manually or automatically of network devices. Various tests can be run as well as diagnostic information can be retrieved on the network devices to detect any current or future defects [40].

Network Management Center The NMC is similar to the OMC but it is not restricted to only regional GSM entities as it is in charge of the all GSM entities in the network. The NMC provides traffic management for the global network and also monitors high priority alarms such as overloaded or failed network nodes. It is usually used in long term planning of a network, but it has the capability to perform certain OMC functions when an OMC is not staffed.

2.4 GSM Network problems

2.5 Summary

Chapter 3

The Frequency Assignment Problem

3.1 Introduction

The Frequency Assignment Problem (FAP)[1] is a generalisation of the graph-colouring problem and is subsequently an NP-hard problem. This is because one has a finite amount of frequencies which needs to be assigned to antennae/transceivers (TRX's) where the amount of transceivers to be assigned frequencies greatly out weigh the amount of available frequencies.

It is inevitable that a network will have interference and we can thus only minimise the amount of interference that might occur on the network - an optimisation problem. Using exact algorithms to find a solution is not practical since the time to find a solution will be polynomial. Generally Metaheuristic algorithms are used to find optimal solutions to NP-hard problems [36]. We will discuss Metaheuristic algorithms which are generally used to find solutions for NP-hard problems in Chapter 4.

A contributing factor to the difficulty of the FAP is due to the scarcity of usable frequencies in the radio spectrum, which forces network operators to reuse their allocated/licensed frequencies in their respective networks. The scarcity of the usable frequencies in the spectrum can be attributed to the overuse of certain bands as well as large scale reuse of frequencies in networks. This has put strain on the spectrum and has complicated the

[1] Also known as Automatic Frequency Planning (AFP) or Channel Assignment Problem (CAP) [36]

management of networks significantly because interference is more likely to occur.

Frequency assignment is the last step in a long process of network setup. Before frequencies are assigned base stations need to be placed and need to be configured, which include azimuth and tilt of the antenna for optimal network coverage. After the base stations are configured they need to be allocated a certain amount of transmitters to achieve a target network capacity [14].

Frequency assignment is only a means to achieve the targeted network coverage as well as network capacity.

In this section we gave a brief introduction as to what the Frequency Assignment Problem is and why it occurs as well as why it is a problem. In the next section we will describe the types of Frequency Assignment in use today and we will also discuss some of the varaints of FAP and also formally define which of the variants we will concentrate on.

3.2 Frequency Assignment Types

In this section we will discuss the different methods that exist to allocate frequencies to cells in a cellular network. We will also state which method we will use through out this paper.

Within the FAP domain there exists different types of the FAP which have emerged over the years as the domain and requirements have changed. We will discuss these FAP variants in section 3.4.

There are a variety of FAP in the domain (which will be discussed in later sections of this chapter) but most of these problems can be classified into two categories based on the assignment scheme they use:

(a) *Fixed Frequency/Channel Assignment* (FFA/FCA) is the process of permanently assigning frequencies to cells (cellular towers). The frequencies assigned are fixed and cannot be changed on the fly while the network is active , since the frequencies assigned to the cell form part of a delicate frequency plan designed to keep interference to minimum.

(b) *Dynamic Frequency/Channel Assignment* (DFA/DCA) is the process of allocating channels to cells as they require it to meet the current traffic demand imposed on them by clients.

Each cell can be assigned multiple frequencies based on the amount of transmitters or TRX's it has. The amount of TRX's in a cell depends on the expected amount of traffic the particular cell must handle.

Most of the research in the FAP has concentrated on the FFA. The reason for this is because FFA is a static technique, which allows it to come up with a better solution since it has more time for calculation. FFA is also easier to implement in practise and allows the network operators to cater for the worst case scenario - heavy traffic load on the network.

The DFA is at the moment a very hard problem because the network frequency plan is constantly changing, which means as the traffic on the network increases the longer the DFA focused algorithm will take to allocate a frequency. This increase in processing time is because the algorithm has to take into account more constraints with a lower available frequency pool. DFA must do this process within seconds since a cell needs to serve clients.

Most researchers have concentrated on solving the FFA using heuristic approaches like neural networks, local search techniques and more recently meta heuristic approaches which include genetic algorithms, simulated annealing , ant colony optimisation and particle swarm optimisation.

In this section we have given a description of the Frequency Assignment Problem and introduced some concepts which we will use throughout the dissertation. In the next section we will present the Mathematics that govern the Frequency Assignment Problem.

3.3 Interference

In this section our disuccsion will focus on Interference. We will describe what interference is and why it is important for cellular networks as well as give an overview of when interference occurs.

Interference occurs when frequencies assigned to connections differ by a small margin. The amount of inference on a connection defines the *Quality of Service* (QoS). One can naturally make the deduction that the more frequencies differ used on connections in a area, the better quality of service one will experience in that area.

Cellular networks use the amount of interference on their networks as qualitative measure for their QoS. A network with high interference would experience a lot of dropped connections/calls, which occurs when the interference is too high to sustain a connection or call for communication, consequently their QoS degrades as interfernece increases.

In the literature a variety of methods are given to calculate the amount of interference in a network. These methods range from theoretical approaches to precise measurements. Regardless of what method is used the end result which they all produce is called an *Interference Matrix* [36].

An Interference Matrix concists of a number of cell pairs (*i,j*), where i is the cell receiving interference and j the cell whose allocated channel is providing the interence. Each cell pair in the matrix has two corresponding values that indicate the level of interference if the *Electromagnetic constraints* are violated [2, 15, 36]. These values are usually normalized to be between 0.0 and 1.0 [14].

Primarily Interference occurs when the Electromagnetic constraints are violated, which are defined as:

Co-Channel — When a cell i and a cell j operate on the same frequency or channel interference will occur [2, 15, 21, 36, 64]. This is called co-channel interference. This constraint is the most important constraint that must not be violated to ensure proper performance and reliability of a modren cellular network [21].

Adjacent Channel — When a cell i and a cell j operate on adjacent channels, their allocated frequencies differ by one i.e. cell i operates on channel f then if cell j operates on either channel *f - 1* or *f + 1* then adjacent channel interference will occur [2, 15, 21, 36, 64]

Co-Site — If cell i and cell j are located at the same site, then their allocated frequency ranges must differ by a certain distance in the frequency domain. This distance is known as the reuse distance [22, 84].

A fourth constraint, kown as the Handover constraint, is also applicable in Cellular networks. This constraint imposes a separation in frequencies when one cell hands over a call to another cell. If this constrain is violated a mobile subscriber will experience a dropped call since the handover between cells fials.The above constraints only account for factors that are in our control.

Interference also occurs due to techincal limitations, natural phenonema and other external factors like other systems. Thus another constraint is imposed on the frequency that is allocated to cell. This constraint is known as the *separation* constraint which imposes a minimum separation between frequencies assigned to a cell [15, 64]. To avoid clashes with other operator frequencies each cell may also have a set of locally forbidden frequencies which are not allowed to be used under any circumstance.

In this section we described what interfenece is and what the consiquences are of too much interfence in a network. We also laid out under which circumstances interference can occur in a wireless network. In the next section we will give a brief overview of in which industries the Frequency Assignment Problem is applicable.

3.4 FAP in the industry

In this section we will list some of the industries where the FAP is encountered. We provide a brief overview how the problem differs compared to other industries. We will also give some references to literature where the FAP and the particular domain are discussed.

3.4.1 Satelite communication

The FAP in the Satelite communication domain occurs with respect to the ground terminals that transmit and receive signals via a satelite. One would assume that the problem includes the satelite, but the problem is only concerned with the frequencies that the ground terminals use. It is interesting to note that the ground terminals can be a base station or a handheld device (e.g GPS or Satelite phone).

In Satelite communication, a signal is transmitted to one or more satelites via an uplink from a ground terminal. The signal is received by the recipient statelites and relayed to the interested ground terminals who receive the signals via a downlink.

The frequencies used by the ground terminals for uplink and downlink communication are separated by a large distance in the frequency domain - the typical distance is much larger than the bandwidth. When frequencies are assigned to transmitters, downlink transmitters are ignored and only uplink transmitters are considered [2].

A radical difference with regard to the use of frequencies compared to the standard FAP in Cellular Networks is that, frequencies are only allowed to be used once. This is sepecific to the satelite domain to avoid interference [2].

There isn't a lot of research on the FAP in Satelite Communication. One of the few recent papers on FAP in this domain is a paper by Lui et al. [35] where the authors employ a Chaotic Neural Network to minimize the interference in a satelite communication system with which they achieve very good results which in most benchmarks finds the global optimum.

Another papers conetrating on this domain is a paper by Houssin et al. [25] where the allocation of frquencies assigned in the satelite system is optimised using Space Division Multiple Access (SDMA). SDMA was developed for use in 3G networks and forms part of the Multiple Access family of techniques (CDMA,TDMA) that are in use in wireless networks today.

It is interesting to note that the authors concentrate on optimising the amount of users served by the system and not interference incurred by the allocated frequencies.

3.4.2 Wireless mesh networks and WLANs

Wireless mesh networks and WLANs[2] are the most recent applications where the FAP is encountered.

Multiple WLANs are increasingly being used to provide backbone support for large fixed line netwroks, enterprise networks, campuses and metropolitan areas. To be able to provide backbone support for these networks, a primary design goal when desiging and deploying these networks is capacity. A limiting factor for WLAN capacity is interference which affects multihop hop settings. Thus the overall network interference needs to be minimized to increase the capacity of the network [70].

Most wireless networks operate on the IEEE 802.11 a/b/g standard. A IEEE 802.11n standard is available but hasn't been finalized yet, even though one can already find wireless hardware operating on this standard. According to the IEEE 802.11g standard only 13 Frequencies are available for use and in some geographical areas a futher limiting constraint is imposed which only allows a certain subset of frequencies to be used [2].

[2]Both applications use the same standard and encounter similar problems in their respective domains.

Typical approaches allocating frequencies include using DCA and FCA[3]. DCA isn't very popular since the dynamic switching of channels lowers the response time on commidity hardware since there is a delay in miliseconds when switching channels. Typical packet transmission times are in microseconds. To garuntee uptime and high responsiveness, FCA is the preferred approach [70].

The FAP in Wireless Mesh Netowkrs and WLANs differ to the standard problem in that it introduces an extra constriant. Channels assigned to links on a node cannot be more than the available interfaces on that particular node. This constraint is known as the *interface constraint* [70]. Another aspect to consider is the placement of access points (AP) in the network, which is similar to the problem cellular networks face with regard to base station placement [2].

There is a wealth of literature on Wireless Mesh Networks. The paper by Subramanian et al. [70] formulate a lower bound using semi-definite techniques and linear programming. Using these lower bounds with their discussed algorithms they get very promising results on a simulation benchmark (ns2). Their discussed solution imposes no specific hardware or topology changes in the wireless network.

Another interesting paper by Chen et al. [11] the authors follow a different route than traditional proposed algorithms with regard to interference. The authors present algorithms that focus on the interference as perceived by the user and not the AP. They also use site specific knowledge provided by Blueprints, Google Earth and Google Maps in their algorithms to predict potential path loss when allocating a frequency to a AP.

3.4.3 Military field communication

In a Military context the FAP is a very difficult problem to be solved due to its dynamic nature. During deployment connections need to be established rapidly between nodes with not gauruntee that the nodes would stay static at locations. Usually nodes are military field phones can be any transceiver device.

Due to the nature of the problem the DCA scheme is used to allocate frequencies to nodes. The Military FAP differs due to the property that any

[3]Discussed in section 3.2 on page 22

of the nodes are mobile and can move at any moment to a new location, potentially interfering with another connection. Two frequencies need to be assigned to each connection that is established, one for each direction of communication. These allocated frequencies must also differ by a certain distance in the frequency domain to prohibit alternating directions of communication interfering.

A lot of literature can be found on Military field communication. This is due to two organizations CELAR[4] and EUCLID[5] making data available to various research groups and allowing them to develop algorithms for frequency assignment.

A comprehensive study by Dupont et al. [13] on the 36 instances of real life data obtained from CELAR. The Authors state that the CELAR data actually has 3 subproblems which occurs in seperate stages. In the first stage a Constrained Satisfaction Problem is encountered when assigning initial frequencies. The second problem occurs when new links are established and frequencies need to be assigned, this is known as the second stage. The last and final stage occurs when a new link cannot be assigned a frequency and thus a repiar is needed. For each stage the authors developed algorithms to try and optimily solve it to produce and overall optimal solution.

For the instances made available by EUCLID the study by Aardal et al. [1] provides results from various groups who worked on the instances provided, which is known as the CALMA project. Various optimization and approximations algorithms were implemented by the research groups and new lower bounds were also found. The authors present each algorithm implemented by the underlying research group. Results are shown for Minimum Interference problems as well as Minimum Span problems.

3.4.4 Television and Radio Broadcasting

The FAP encountered in broadcasting very much resembles the problem domain found in Cellular networks. The only notable difference is the required distance allocated frequencies must differ in the frequency domain is larger in broadcasting than in cellular networks [2].

Since the problem resembles the problem found in Cellular networks, there are few articles that specifically discuss frequency assignment in broad-

[4] Centre Electronique de L'Armement

[5] European Cooperation on the Long Term Defense

casting as a main topic. A paper that does specifically discuss FAP in broadcasting is presented by Idoumghar and Schott [28]. The authors present a distributed hybrid genetic algorithm and a cooperative distributed tabu search algorithm. They compare these algorithms with their sequential counterparts of their algorithms and with a ANTS algorithm. The benchmark instances they used were provided by the TDF-C2R Broadcasting and Wireless research center.

3.4.5 Cellular Communication

Cellular communication[6] can be considered the main driving force with regard to research in the Frequency Assignment domain. As new standards are developed and used in 3G networks, in general a frequency assignment problem still needs to be solved since these techniques do not eliminate interference entirely, but the do make it *less* likely to occur.

One such technique that is mostly used in GSM networks is called Frequency Hopping, which as the names emplies, the transmitters "hops" onto different frequencies according to a predefined sequence of frequencies. The frequency can change per packet if the underlying hardware can handle it otherwise it switches per connection [2, 15, 46].

The FAP in the Cellular domain is the most researched topic and is considered the default domain of the problem. As such, most of the literature concentrates on this domain and one can find a lot of articles in the literature presenting viable algorithms that produce real world solutions [15].

Because the problem is NP-Hard most presented algorithms are either of the metaheuristic type or more recently of the swarm intelligence type. Both of these algorithmic types are discussed in Chapter 3 and 4 respectively.

Besides optimizing algorithms, there is also a wealth of literature on upper and lower bounds for the FAP. Using lower bounds in FAP orientated algorithms can produce very favourable results as demonstrated in the paper presented by Montemanni and Smith [47]. Using a lower bound in conjunction with their algorithm they achieved a new optimum in a variety of benchmarks, most notably in the COST 259 benchmark.

Other papers in the literature contribute by providing different modeling techniques such as the paper by Borndrfer et al. [8]. The interested reader

[6] An overview of a Cellular Communication technology called GSM is presented in Chapter 2.

who wants to know more about the problem domain, general modeling techniques used as well as the most common algorithms used in directed to the study by Aardall et al. [2].

In this section we discussed the different industries in which the Frequency Assignment Problem is encountered and also gave a brief description on how the problems are different with regard to what constraint it imposes on the frequencies for the particular industry. We also gave some brief references to some literature where the FAP is discuess with regard to the particular domain. In the next section we will give a brief description of the different types of Frequency Assignment Problems as well as give a small discussion on the literature found for the individual problems.

3.5 Frequency Assignment Problem types

I We will start of giving a brief overview on one of the first and oldest problems in the domain and we will end of discussing the problem we will base our implementation on.n this section we will give a brief explanation on some of the various problems that exist for the FAP. We will start of giving a brief overview on one of the first and oldest problems in the domain and we will end of discussing the problem we will base our implementation on.

3.5.1 Minimum Order FAP

The Minimum Order FAP (MO-FAP) was the first FAP that emerged in the 70's. The MO-FAP is concerned with assigning frequencies to transmitters while interference is minimized as well as minimizing the amount of different frequencies that are used.

In MO-FAP frequency re-use is prioritised and the usage of a frequency has a certain cost associated with it. The reason for this is because when the wireless network industry started out, operators were billed according to the amount of different frequencies they used. In the beginning frequencies weren't cheap since they were sold per unit [2, 46].

Over the years as the law governing the wireless spectrum changed and new technology as well standards emerged, thus MO-FAP has lost its relevancy. Companies aren't billed according to the different frequencies they use, but they purchase licenses from a regulatory body. This license usually stipulates what frequency band the network is allowed to use.

In Some instances a certain band of frequencies is put up for auction by a regulatory body, to which interested parties can bid to own the specified spectrum. Due to the shift in how frequencies are allocated to network, neither the regulatory bodies nor the network operators care about the amount of different frequencies are used. Thus MO-FAP has lost its relevancy in the modern wireless industry.

3.5.2 Minimum Span FAP

The Minimum Span FAP (MS-FAP) is a problem that is very relevant today, especially when network operators want to deploy a new network in a region. The MS-FAP is concerned with keeping the interference below a certain level during assignment as well as minimizing the span. The interference threshold used, is specifed by the network designer as the minimum allowable interference on the network.

The span is defined as an interval on the frequency domain. This interval is calculated by taking the difference of the maximum and minimum frequnecies used during assignment. With the span value, network operators are able to request certain frequency bands and know their network will be able to operate at suitable interference levels [2, 46, 63].

The MS-FAP and MO-FAP are two very similar problems, the only difference is that MO-FAP focuses on minimizing different frequencies and MS-FAP forcuses on minimizing the interval of frequencies used during assignment [2]. The Philadelphia benchmark is usually used to gauge how good the algorithm performs.

3.5.3 Minimum Interference FAP

The Minimum Interference FAP (MI-FAP) or Fixed Spectrum FAP (FS-FAP) is typically encountered after the network operator has obtained a frequency band from a regulatory body. Other problems use matrices to forbid certain frequencies from being with certain transmitter [2, 15, 19, 46].

Unlike previous discussed problems, in MI-FAP any available frequency in the allocated band may be used even though it produces interference. The other problems are concerned with the frequencies used, even though they might be violating some constraints that incur a huge amount of interference. The interference value doesn't play a large role in their respective objective functions. In MI-FAP the objective is to minimize the total amount of interference on the network. It is important to note that this amount of interfernece might not necessarily be zero [2, 15, 19, 46].

The MI-FAP is the most encountered problem currently in cellular networks, since there are more operating networks than new networks being designed in the cellular industry today. This particular problem for form the focus for this dissertation.

Since MI-FAP is very close to real world instance problems, authors tend to use real world instances or benchmarks that resemble real world instance to test the quality and efficiency of their algorithms [2, 15, 19, 46]. We'll benchmark the quality and efficiency of our solution with the COST 259 benchmark which is discussed in section 3.7.

In this section we laid out the different types of Frequency Assignment Problems there are in the literature. We also gave a brief discussion on some of the literature found on the individual problems. Finally we formally stated on which one of the frequency assignment problems we will be concentrating on, namely the Fixed Spectrum Minimum Interference Frequency Assignment Problem (MI-FAP).

In the next section we will give a Mathematical definition for the Fixed Spectrum MI-FAP which will form the bases for the objective/cost function that we are going to minimize to find an optimal frequency plan.

3.6 Fixed Spectrum MI-FAP Mathematical Formulation

In this section we will give a Mathematical definition of the Frequency Assignment Problem which will form the core of what our algorithm discussed in this dissertation will optimize. We'll start of by denoting the symbols we will use and then we will give the Mathematical definition of the cost function we will minimize.

The Frequency Assignment Problem can be represented as a graph colouring problem hence it is known to be NP-Complete. Before we can formally define the Frequency Assignment Problem we first need to introduce some symbol definitions.

$$G = (V, E) \tag{3.1}$$

$$V = \{v_0, v_1, ..., v_i\} | i \in \mathbb{N} \tag{3.2}$$

$$E = \{v_0v_1, v_0v_2, ..., v_iv_j\} | v \in V, \forall ij \in \mathbb{N}, i \neq j \tag{3.3}$$

$$D = \{d_{01}, d_{02}, ..., d_{ij}\} | \forall \{i, j\} \in E, \exists d_{ij} \in \mathbb{N}^+ \tag{3.4}$$

$$P = \{\{\bar{p}_{00}, \bar{\bar{p}}_{01}\}, \{\bar{p}_{10}, \bar{\bar{p}}_{11}\}, \ldots, \bar{p}_{i0}, \bar{\bar{p}}_{i1}\}\} | \forall \{i, j\} \in E, \exists p_{ij} \in \mathbb{N}^+ \tag{3.5}$$

$$F = \{0, 1, 2, 3, ..., k\} | \forall k \in \mathbb{N}, \forall v \in V \exists f \in F \tag{3.6}$$

$$d_{ij} < |f(i) - f(j)|, \forall ij \in \mathbb{N}, i \neq j \tag{3.7}$$

Let G (see 3.1) be a weighted undirected graph, where V (see 3.2) is a set of vertices. Each $v \in V(G)$ represents a transmitter in the frequency assignment problem.

E (see 3.3) is a set of edges. An edge consists of two vertices v_i and v_j that are joined because there exists a constraint on the frequencies that can be assigned between the two vertices or transmitters. Each edge has two associated labels d_{ij} and p_{ij} [8, 47].

The label d_{ij} that is part of the set D (see 3.4) denotes the maximum separation that is requried to exists between frequencies assigned to two transmitters v_i and v_j. Using $f(i)$ to denote the frequency assigned to i, we can determine using equation 3.7 if the interference involving the transmitters v_i and v_j is acceptable [8, 47]

The other label, p_{ij}, forms part of the set P (see 3.5) which is refered to as the Interference Matrix[7]. Each label p_{ij} contains two values which represents interference[8]:

- $\bar{p}_{i0}$ represents the value for co-channel interference [8, 47].

- $\bar{\bar{p}}_{i1}$ represents the value for adjacent channel interference [8, 47].

Lastly we have the set F (see 3.6) that denotes a set of consecutive frequencies for every transmitter in V [8, 47].

Formally the Fixed Spectrum Frequency Assignment Problem (FS-FAP) can now be defined as a 5-tuple $FS-FAP = \{V, E, D, P, F\}$ with a required mapping of $f : V \rightarrow F$ [47]. The objective of the FS-FAP is to find an

[7]Discussed in section 3.3 page 23

[8]Interference values can be zero in some cases

assignment of frequencies to transmitters that minimizes the sum of total interference (see 3.9).

$$c(p_i) = \begin{cases} \bar{p_{i0}} & , \text{if } |f(i) - f(j)| = 0 \\ \bar{\bar{p_{i1}}} & , \text{if } |f(i) - f(j)| \leqslant d_{ij} \\ 0 & , \text{if } |f(i) - f(j)| > d_{ij} \end{cases} \tag{3.8}$$

$$TotalInterference = \sum_{i=0}^{\mathbb{P}} c(p_i), p \in P \tag{3.9}$$

In this section we Mathematically defined the Frequency Assignment Problem using the symbols we defined. In the next section we will give a brief discussion on the different Frequency Assignment Benchmark Problems that exist and also define the benchmark we will be using in our implemention.

3.7 FAP Benchmarks

In this sections will discuss some of the most used benchmarks in the FAP domain. We will start of with the first benchmark that was introduced in the 70s and end of with a disuccsion on the benchmark we will be using to test our implementation.

3.7.1 Philedelphia Benchmark

The Philedelphia benchmarks are derived from an instance that was introduced in 1973 by Anderson. Each instance is a hexogonal grid of cells that overlaps the area of interest. At the center of each cellthere is a transmitter. Past approaches used these hexagonal systems to model modern cellular networks [2, 39].

In this benchmark interference is measured by a co-channel reuse distance. This distance stipulates that the difference of the frequencies assigned to two cells must greater of equal to a certain value d. A frequency cannot be assigned to a cell if it violates this minimum distance [2, 39].

These benchmarks are typically used to test algorithms developed for MS-FAP, since there is no concept of cost or penalty for interference incurred by violating constraints.

3.7.2 CELAR

In 1994 EUCLID introduced a project called CALMA which was a combined effort by various Europena goverments that were part of EUCLID to investigate algorithms for Military applications. The project was granted to six research groups. Within the project 36 instances were made available by CELAR for Radio Link Frequency Assignment [2, 13].

All the CELAR instances have the constraint that the difference between frequencies assigned to interfering radio links must be greater that a certain predefined distance in the frequency domain. This is a soft constraint and may be violated. Another constraint in the CELAR instances is that each pair of parallel links must differ by an exact predefined distance. This constraint is a hard constraint an may not be violated [13].

These instances were initially not available to the general public as it was contained to be within the CALMA project. In 2001 the CELAR launched an the International ROADEF challenge, were certain instances from the CALMA project were made available for the research teams taking part in the challenge. The instances made available had been modified to take polarizations and controlled relaxations of certain EMC constraints [24].

3.7.3 COST 256

The COST (COoperation europene dans le domaine de la recherche Scientifique et Technique) 259 is a set of real world GSM instances made available by die European Union. The instances are publicly available and can be downloaded for free at http://fap.zib.de/ (FAP Web 2007). The website also constains the most recent results obtained by researchers using these instances [2, 15].

The instances are fairly difficult due to the large amount of transmitters (900 - 4000) that need to be assigned frequencies, with a relatively small amount spectrum of frequencies. The main important characteristic of this benchmark is that is resembles real world GSM network data, which is why we the authors have selected this as the primary benchmark we will be concentrating on [2, 26].

More specifically we will concentrate on a small subset of the instances that are available, namely Siemens1, Siemens2, Siemens3 and Siemens4. In the paper by Montemanni and Smith [47] the same subset of problems is used and to date their algorithm has produced some of the best results.

3.8 Summary

Chapter 4

Heuristics Algorithms

4.1 Introduction

Meta-heuristics is a sub domain of the artificial intelligence domain. It evolved out of a need for more efficient search techniques with regard to hard problems.

Meta-heuristics forms part of a collective body of algorithms that use heuristics to search a particular domain's problem space, for the most optimal solution adhering to certain hard and soft constraints. Some of the most important Algorithms that form part of this collective body is:

- Tabu Search
- Simulated Annealing
- Genetic Algorithm

The above mentioned algorithms aren't the only algorithms to form part of this sub-domain, but they are the algorithms that have received the most attention in the literature and generally produce good results [42].

In this chapter our main focus will be to discuss each of the above listed algorithms. We will start of by briefly discussing the characteristics of meta-heuristic algorithms after which we will discuss each of the above algorithms in detail. We will also provide a literature study for each algorithm in order for us to see how an algorithm needs to be changed and optimised for a particular problem domain.

4.2 Characteristics of Meta-heuristics

NP-Complete problems have been proven to not be solvable in polynomial time by traditional search methods such as A* search, Breath First Search and Depth-First Search. Meta-heuristic algorithms on the other hand are much more efficient in searching the problem space and produce much better results in a short amount of time.

These algorithms are considered to be *general-purpose* algorithms and can thus be applied to a wide variety of optimization problems with only small modifications that need to made to the algorithm model [37].

Meta-heuristic Algorithms do not search statically by testing and evaluating every possible permutation in the solution space. Instead these algorithms make use of certain strategies and heuristics (specific to the problem domain) to search the solution space intelligently through trail and error [7].

These algorithms iteratively move through the solution space, using a heuristic to guide the search to move to more desirable regions in the solution space where there is a high probability of obtaining high quality candidate solutions [42, 48].

Meta-heuristic based search methods aren't guaranteed to find the most optimal solutions in the solution space, instead these methods are usually used to find near-optimal solutions. Thus most algorithmic development in the meta-heuristic domain focus developing new techniques that will increase the probability that a good solution will be obtained in difficult combinatorial problems [7].

Similarly, Meta-heuristics aren't guaranteed to find "good" solutions or perform well in each problem domain it is applied. The quality of the solution and performance of the meta-heuristic is very much depended on the expertise of the algorithm designer [83].

The standard meta-heuristic algorithms won't take advantage of specific domain knowledge to exploit the search domain and will produce relatively poor results. It is up to the algorithm designer to modify the algorithm sufficiently based on domain knowledge he/she as obtained [83].

Although heuristics play a key role in the performance of meta-heuristic algorithms, it isn't the only factor that has an impact on performance and results. Algorithms also use techniques and concepts from other system paradigms like multi-agent systems.

In multi-agent systems, multiple agents have to communicate with each other and the system as a whole has to perform some sort of autonomous self-organization. This social and self-organization concepts enable these systems to be distributed,robust and flexible. Which is why in meta-heuristic algorithms that are population-based,hybrid and/or distributed these same concepts are used to better exploit the solution space [41].

Meta-heuristics tend to slowly converge on an optimal solution, hence wasting valuable computing cycles. Therefore, a recent trend in research using meta-heuristics for problem solving often pair the algorithms with local search methods to increase the convergence rate of the algorithm to obtain a solution faster [23].

In this section we introduced the characteristics of meta-heuristics which sets these algorithms apart from the conventional algorithms used on difficult problems. We gave a general overview on how solutions are obtained as well as the quality of solutions. We also briefly discussed why for each problem domain the algorithm used, must be changed to fit the domain.

In the next section of this chapter we will discuss the Tabu Search meta-heuristic which we are investigating.

4.3 Tabu Search

4.3.1 Introduction

Tabu Search (TS) was first proposed by Glover as a new searching technique to help algorithms avoid getting stuck in local optima present in combinatorial and optimization problems [60]. Since Glover introduced the algorithm in the 1980's, Tabu Search has been applied to a wide range of problems that include a wide variety of problems such as the Vehicle Routing Problem, Frequency assignment Problem, Capacitated-Lot Sizing Problem, Nurse Scheduling and the Resource Constrained Assignment Problem. Even though the problems mentioned differ by a large margin, the algorithm has been relatively successful in most of optimization problems it has been applied to. If we observe the results presented in the following research [9, 18, 45, 47, 53, 54, 60, 69, 72, 79] we can deduce that Tabu Search has on average obtained the best results compared to previous attempts with other algorithms.

Tabu search resembles in its most basic form the Hill-climbing search algorithm, but it differs in the sense that Tabu Search keeps a memory of its recent moves in the solution space [72].

General search algorithms like Hill-climbing, Random-restart or Scatter search tend to get stuck on local optima. The local optima might be a very attractive solution and thus general search algorithms will not move to better solutions since according the algorithms built in strategy it has found the best solution, but in actual fact its solution is the best it its *local* search space but not in the *global* search space. This is why an important characteristic that algorithms being applied on optimisation problems need to posses is breaking out of local optima.

In the next section we will explain what makes Tabu Search such a better algorithm that previous algorithms and why it is able on average to produce better results.

4.3.2 Important Tabu Search characteristics

In this section we will discuss some of the key characteristics and techniques that Tabu search exhibits that enables it to find relatively good solutions in a short amount of time. We will start of briefly discussing how the start solution Tabu Search iteratively improves upon. After initial solution generation we will give an overview of research done on neighbourhood strategies for TS. One of the most important features of TS will be discussed in the memory structures section. Finally, we will finish of this section with a discussion on the two search phases present in TS.

Initial Solution Generation

The core feature of the TS algorithm is sequentially improving the initial solution [86]. Thus an important consideration to make is how initial solutions are generated for the TS to start on. Random initial solutions might seem to be a good starting point, but by introducing randomization it becomes hard to control the quality of the end solution. Hence the generation of starting solutions must be controlled to limit the infeasibility of potential solutions [86].

Neighbourhood search

Tabu Search uses a neighbourhood local search process to explore the solution space. There is no set process of how neighbourhood candidate solutions are selected. Depending on the problem the TS is applied different neighbourhood solution selection strategies are needed. The overall quality of the solution produced by TS is also dependent on the neighbourhood search strategy used [86].

The TS algorithm isn't limited to just one neighbourhood search strategy. In the paper by Gopalakrishnan et al. [18] five neighbourhood move strategies are developed and are used interchangeably, in some cases a strategy is used three times in a row due to stagnation in the search space. However to combat this stagnation, the authors opted to use all the move strategies 15 percent of the time, and the last four moves strategies for 85 percent of the time when generating neighbourhood solutions.

Other neighbourhood strategies developed is one developed by N. A. Wassan [79]. In the authors paper a neighbourhood selection strategy is used that exchanges route nodes from initial vehicle routes for the Vehicle Routing Problem. This route exchange enables the TS algorithm to search much more broadly due to the constant supply of different solutions. Since initial solutions are constantly modified it enables the TS procedure to be a very fined grained process, because often a small changes in a potential solution can have a big impact on the overall proposed solution by the TS algorithm.

In the research done by Zhang et. al [86] an interesting neighbourhood selection scheme called *dynamic penalty* is discussed. When the algorithm moves onto an infeasible solution a penalty is imposed. By dynamically changing the penalty that is imposed the "feasibility" of solutions produced is influenced. Therefore, when and if the algorithm continually produces infeasible solutions, the penalty imposed is increased as to guide to algorithm to produce more feasible solutions. Finally, in the case when the algorithm is stuck on local optima, the penalty is reduced, which allows the algorithm to consider moving onto infeasible solutions thus escaping local optima.

Considering all the research done to develop new neighbourhood selection strategies that improve Tabu Search to search the solution space more efficiently and produce better faster solutions, Tabu Search still has some draw-backs, especially with problems that have very large solution spaces [6].

Tabu Search is an iterative algorithm, executing a set of operation sequentially until a stopping criterion is met as can be seen in the flow-diagram presented earlier. At each iteration the algorithm has to determine feasibility of the immediate neighbourhood candidate solutions [6, 45]. Therefore each candidate must be evaluated by some function, which may be a costly operation in terms of computational cycles as well as in terms of time. Hence, this constant evaluation can drastically reduce the overall performance of the algorithm, since it spending more time calculating feasibility than actually searching the solution space [6, 45].

Memory structures of Tabu Search

The Hill-climbing and Random-restart algorithms are able to break out of local minima, but there is nothing stopping these algorithms from avoiding the local optima with their second or n-pass in the search space. Tabu Search differs from these algorithms by incorporating an important concept; the notion of memory.

In its most basic form Tabu Search keeps a local memory of all its recent best moves, and puts them into a *Tabu List* that has a predefined size. In the literature the Tabu list is also referred to as the *Tabu tenure* [18, 31, 79, 86]. The algorithm is not allowed to move to any solution that is in the Tabu list unless a solution that is *Tabu* is better than any current moves available in the immediate search neighbourhood [18, 31, 79, 86]. The process of overriding a solutions Tabu status in the Tabu tenure is called the *aspiration criterion* [18, 31, 79, 86]. With the use of the Tabu tenure and the aspiration criterion, the algorithm is able to avoid cycling,local optima as well as searching in a to narrow region [5, 54].

Research done by Ashish Sureka and Peter R. Wurman makes an important distinction with regard to the memory scheme that is used in the TS algorithm. Two memory schemes are discussed; *explicit* memory and *attribute-based memory* [72, 73]. Between the two memory schemes the explicit memory scheme is the most used in the literature [45].

With explicit memory the algorithm stores a complete solution in the Tabu tenure, hence the algorithm is prohibited to move to that position in the solution for as long as the solution is in the Tabu tenure [72, 73]. With attribute-based memory the algorithm stores the *operation* the is used to move from the previous solution, to the current solution [72, 73]. Therefore with attribute-based memory, the Tabu tenure intended function is changed from prohibiting certain solutions already encountered, to rather prohibit making changes to the current solution that would lead to solutions already present in the Tabu tenure [72, 73].

In research conducted by D.M. Jag and G.T. Parks and T. Kipouros and P.J. Clarkson, the authors add two additional memory structures called *Medium Term Memory* (MTM) and *Long Term Memory* (LTM) besides the standard Short Term Memory, typically referred to as the Tabu List [29] . Each additional structure remembers a different set of solutions for use by the diversification and intensification phases in the algorithm. These two phases will be discussed in the next section.

STM purpose is similar to the traditional Tabu list, to store the most recent solutions produced by the algorithm. MTM is designed to remember optimal or near optimal solutions. These solutions are therefore used later in the intensification phase. Finally, the LTM structure stores all the regions that the algorithm has already explored and is thus used in the diversification phase of the algorithm [29].

Search phases

As Tabu Search searches through the solution space, it goes through two cycles of search phases called *diversification* and *intensification* [9,16,23,31].

The diversification phase in the TS algorithm, is the phase where the algorithm is directed to areas in the solution space which hasn't been explored yet. Diversification is usually applied by the algorithm as soon as mechanisms monitoring the memory, notice that solutions being produced are being repeated [16, 79].

In the literature diversification is achieved by new and innovative methods. A diversification strategy developed in research presented by Wassan [79] discusses a strategy called *escape diversification* where the algorithm is taken out of its current position in solution space as soon as solutions are being repeated.

In Research done by Fescioglu-Unver and Kokar [16] a strategy is presented that consists of two components namely the *Observer* and the *Diversifier*. The goal of the Observer is to continually monitor the best solution obtained by the algorithm whether it violates the *stagnation period*. The stagnation period is defined as the amount of iterations where the current best obtained solution hasn't changed [16].

As soon as the stagnation period is exceeded by the algorithm the Observer component activates and transfers the necessary information needed by the Diversifier component. The Diversifier component dynamically changes the size of the Tabu tenure based on the information the Observer gathered. The diversifier mainly targets older moves to diversify, but for short burst of time it would decrease the Tabu list size to a very small value in an attempt to combine new and old moves [16].

The specific mechanism used to define a new position where the algorithm can continue search, should ideally select areas in the solution space which have not been explored yet. Therefore, the diversification phase typically makes extensive use of the knowledge present in the long term memory structures as an indication to what areas of the solution space have been previously explored and which areas have not [9, 16, 23, 31].

Intensification is usually the first phase of the Tabu Search algorithm, since it is responsible to build up a history in memory for which the diversification phase can act upon. Fescioglu-Unver and Kokar also presented a intensification strategy based on control theory in their research [16]. The authors identified the repetition length as a critical value for their intensification strategy to be based upon. The repetition length is a control measure that defines how many times a solution may be repeated.

Repetition length was chosen because the authors observed through experimentation, that as solutions were repeated the algorithm was intensifying around a point in solution space. As the repetition length was increased that algorithm is forced to find more diverse solutions thus moving away from the intensification point [16].

In this sub section we discussed the different phases of the Tabu Search algorithm. We discussed the intended purpose of each phase and presented some relevant research done in this area.

In this section we presented the pseudo code for the Tabu Search algorithm as well as presented a Flow diagram depicted when and why certain phases are activated in the algorithm. We then gave a overview of the core characteristics that are important for the Tabu Search algorithm. In the next section we will discuss Simulated annealing, where will start off by giving the pseudo code and flow diagram for the algorithm and end of the section by giving a overview on the core features of the algorithm.

4.3.3 Algorithm and Data flow

4.4 Simulated Annealing

4.4.1 Introduction

Simulated Annealing (SA) is a heuristic search technique proposed in the 80's by Kirkpatrick to solve combinatorial optimisation problems. The technique is based on a natural process which is known in metallurgical as Annealing [33,52,71,76]. Kirkpatrick was the first to use the simulated annealing to solve optimisations problems but the basic algorithm structure was defined in by Metropolis et. al. in 1953 [51,76]

Annealing is the natural process of crystallization when a solid is heated to a high temperature and then systematically cooled to a low temperature to reach a crystallized form [4,34,43,76]. This crystallized form of the solid is known the be the global minimum of the solids internal energy state. When the solid is rapidly cooled from a high temperature, the molecules have no time to reach a thermodynamic equilibrium stage [4, 34, 43, 76]. Therefore, the molecules of the solid have high energy and the resultant structure has no real crystalline form, thus the solid energy is at a local minima [34,43,76]. When the solid is slowly cooled in a controlled manner, the molecules are able to reach a thermal equilibrium at each temperature [4, 34, 43, 52, 76].

In the algorithm the energy state is the *cost function* that needs to be minimised, and the molecules are the *variables* which represent the solutions, and thus their state needs to be optimised to reach the desired energy state.

The following equation is the standard probability function that is used to determine when an uphill move performed by the algorithm. This function is known in the literature as the *Metropolis Criterion.*

$$M_{AC} = \begin{cases} 1, & \text{if } f(y) \leq f(x) \\ exp(-\frac{\Delta E}{T_k}), & \text{otherwise} \end{cases} \tag{4.1}$$

The function f is the objective function or a function that determines the state of a given position in solution space [82]. The parameter T_k is the temperature of the algorithm at iteration k [82]. Finally, ΔE is the change in "energy" between two solutions x and y [82].

The main purpose of the SA algorithm (like most optimization algorithms) is to minimize or maximise the cost function [71]. This cost function typically evaluates a solutions desirability compared to other solutions in the immediate *neighbourhood* of the algorithms current position [12]. Typically a neighboring solution is only selected as the new best state if its desirability ranks hire than the current solution. When the algorithm moves to a better solution from the previous solution, the move is typically referred in the literature as a *downhill* move [76].

The best state isn't always selected, in some case the algorithm is also able to move to solutions which are worse than the current solution. A worse solution is only selected based on some probability which is controlled by the *annealing temperature* of the algorithm [12]. At a high annealing temperature the probability that the algorithm will select a bad solution is very good. As the annealing temperature decreases so does the probability that a bad solution will be selected [76]. When the algorithm moves to a worse solution, the move is typically in the literature referred to as a *uphill* move [76]. Uphill moves allows the algorithm to breakout of local minima and can lead the algorithm down a different path which may ultimately result in obtaining the global optimum [71].

The SA algorithm is also very popular due to the basic structure of the algorithm being generic [59]. Therefore, applying the algorithm to other problems is relatively trivial since only small changes are required. These changes usually need to applied to the *Neighborhood selection* scheme and the *Cooling Schedule* [59, 75]. Both of these concepts will be discussed in the next sub section.

In this subsection we gave a brief overview of the Simulated Annealing (SA) algorithm. We briefly discussed what the algorithm is based on and how the algorithm goes about searching the solution space. We also introduced some concepts like move probability selection and annealing temperature, which forms part of the core the algorithm. In the next section we will explain some of the concepts we've touched upon in this section as well as more advanced concepts that makes the algorithm unique.

4.4.2 Important Simulated Annealing characteristics

In this section we will provide five brief discussions on the characteristics of the Simulated Annealing Algorithm that we think make the algorithm unique. We will start of with a discussion on Markov Chains. An discussion on one of the most important defining characteristic, namely the cooling schedule, will follow after the Markov Chain section. Furthermore we will discuss the importance of the initial temperature generation and what impact it has on the overall algorithm. Finally, we will end of this section with an overview on the efficiency of the algorithm.

Markov Chain

The SA Algorithm is typically modelled by using Markov chains due to each Markov chain represents a set of trails that the algorithm has executed at the same temperature. It has been proven with the use of Markov Chain theory that SA will find the global minimum in the solution space [38]. This proof is only valid when the following properties for the underlying Markov chain hold [51]:

- It must be irreducible
- It mustn't be periodic
- The detailed balance condition must hold

Due to the above proof the SA algorithm has been applied to a wide range of optimisation problems because as long as the algorithm designer can uphold these properties the find the global optimum. Even though the algorithm will find the global optimum, the algorithm is known to take a very long time to do so.

Cooling Schedule

The Cooling Schedule / Annealing Schedule is the most defining characteristic of the SA algorithm. It is the procedure where the natural annealing process is mimicked. The temperature of the SA algorithm is a control parameter that defines how much the algorithm moves around in the solution space.

In general, when the SA algorithm temperature has a very high value most solutions that are produced from the neighborhood are accepted [38]. Thus the algorithm moves freely in the solution space with little constraints. As the temperature decreases the probability that the algorithm will select bad or just any solution gets lower. When the temperature is very low, the SA algorithm is similar to a greedy algorithm in a sense that it only accepts downhill movements [38].

In the literature there are three annealing schedules in common use are namely *the logarithmic schedule*, the *geometric schedule* and the *Cauchy schedule* [51, 71].

The standard and most common used schedule is commonly known as the logarithmic schedule and is based on Boltzmann annealing [51]. The main disadvantage of this schedule is that is slow due to its logarithmic nature [51]. It also requires that moves be generated from a Gaussian distribution for it to be able to reach the global minimum [71]. The logarithmic annealing function has the following form:

$$T_k = \frac{T_0}{ln(k)}, \text{where k is the iteration value} \tag{4.2}$$

The Cauchy schedule is faster than the logarithmic schedule. Similar to the logarithmic, this schedule also has a movement requirement. Moves must be generated from a Cauchy distribution for the algorithm to be able to reach the global minimum [51, 71]. The Cauchy schedule is also typically referred to as Fast annealing [51]. The schedule has the following form:

$$T_k = \frac{T_0}{k} \tag{4.3}$$

Finally, the fastest annealing schedule is known as the geometric or exponential annealing schedule [71]. This schedule introduces the concept of *re-annealing*. Re-annealing is a procedure by which all SA temperatures are rescaled [51]. This schedule has no move generation requirement to reach the global minimum, since there is no regious proof in the literature to prove it [71]. The geometric schedule has the following form:

$$T(i) = T_0 exp(-C_i), \text{where C is a constant} \tag{4.4}$$

Initial temperature

The initial temperature is a very important parameter to define in the SA algorithm, since it defines a point from which the cooling schedule will start of from. Therefore, depending on what the initial value of the temperature is the final result that the algorithm will produce can be influenced [59, 67, 81].

When the initial temperature is set to a very high value the algorithm takes a long time reach a result. On the other hand if the initial temperature is set to a very low temperature the algorithm might converge to quickly and thus produce a result which may be the local minima [59, 67, 81].

The initial temperature together with the cooling factor allows the algorithm designer to defines the time window for the algorithm to escape local minima, as well as the rate of convergence to an optimum solution [59, 81].

A low initial temperature together with a small cooling factor makes the time window for the algorithm to leave a local optimum very small [81]. With a high initial temperature and cooling factor value that is almost 1, the time window for the algorithm to leave the local optimum is much larger [81].

When the algorithm is near a global optimum, a low initial temperature and low cooling factor will allow the algorithm to reach the optimum faster in the solution space. In contrast, if a high temperature and a very low cooling factor is used the algorithm will take longer to reach the optimum even tough it is near the global optimum [81].

Move generation

Most of the research done on the SA algorithm focuses on the annealing schedule and not so much on the move/solution/neighbourhood generation. Typically an initial solution is generated and then small changes are made to the solution to represent a new solution. The solution is said the be perturbed to the next solution.

In research done by Tseung and Lin [76] a initial solution isn't modified but a move generation technique known as *Pattern* search is used. Pattern search has two forms of movement namely the exploratory move and the patter move. The exploratory move continually changes the certain variables of a solution [76]. This is done so that it can rapidly find and identify a "downhill" move. The Pattern move uses the information gathered by the exploratory move to move towards the minimum of the function [76].

Algorithm efficiency

The algorithm is also efficient with regards to CPU cycles when compared to the Genetic Algorithm because it only has to evaluate a certain number of moves each iteration, instead of a evaluating a whole population each iteration. Unlike Tabu Search the basic SA algorithm does not keep any memory and is therefore memory efficient, but in contrast suffers the risk that solution will cycling. Which is hwy the number of iterations spent at a temperature, since the longer the algorithm spends at a temperature the higher the probability is that solutions will cycle.

4.4.3 Algorithm and Data flow

In this section we will give the start of by providing the pseudo code of the Genetic Algorithm. We will also provide a flow diagram depicting the general search procedure flow of the algorithm.

4.5 Genetic Algorithm

4.5.1 Introduction

Genetic Algorithm (GA) is a stochastic search method which is based on the natural process of genetic evolution and the Darwinian concept of "survival of the fittest" [17,20,32,77]. GA was initially developed for adaptive systems by Holland but has then, been widely used in the optimization field of study due to its effective exploration of the solution space as well as its relative success in multi-dimensional problems [17, 77, 78]. The wide use of the GA algorithm can also be attributed to its generic algorithm structure as well as the ease of implementation of the algorithm [32, 77]. GA are application dependant and thus the designer needs to tailor the algorithm to his needs to obtain good results [20].

The GA search procedure involves searching the solution space through artificial evolution and natural selection [30, 66, 77]. An individual or point in the solution space is known as *chromosome* in the literature [10]. An initial set of chromosomes (referred to in the research as the *population*), are randomly generated [20, 30, 66, 77]. Unlike other algorithms, GA does not concentrate on one point when searching the solution space, but concentrates on a wide range of points represented by the population [17, 30, 77]. Each chromosome in the solution space represents a string encoding of the problem parameters [77]. Encoding problem parameters also attributes to the wide use of the GA since difficult mathematical problems can now be easily modelled [20].

The population is artificially evolved each iteration by using a set of stochastic operators [74]. This set consists of a selection operator, crossover operator and mutation operator [66, 74]. Each operator plays an important role in emulating the evolutionary process. The selection operator is in charge of applying the *objective function* to each chromosome in the population [10,32]. Depending on if the selection operator is setup for maximization of minimization, each individual is ranked based on its "fitness" or objective function value. In accordance with the Darwinian theory, only the fittest individuals are selected from the population [10]. The fittest individuals are copied and sent to the the next phase of the algorithm which is known as the *Reproduction* phase [10].

Depending on how complicated the objective function is and how large the population is, the selection phase may be the most computationally expensive as well as time consuming [20]. The Reproduction phase mostly consists of string manipulations on the chromosome, to drive to search forward and it thus a phase which is completed quickly, especially if the string encoding is of a binary nature [20, 32]. These string manipulations occur through the application of the crossover and mutation operators [62]. The reproduction phase is where a new population is generated for the next generation to be evaluated by the selection operator. The reproduction is said to generate "offspring" from the selected fittest population who are in turn known as the "parents" of the offspring [10, 62]. Reproduction will occur until a certain number of predefined generations is reach or a suitable solution is found to be greater/less than a certain fitness threshold that would indicate a good chromosome [58].

There are two basic forms of the GA where both forms differ in the way that offspring and parents are handled [77]. The one form is called the *generational* GA and the other form is known as the *steady-state* GA [77,80]. With the generational GA the offspring aren't immediately used in the next generation, instead they are kept in a pool until the pool reaches a certain size [77]. The offspring are then used to replace the parents entirely in the next generation [77]. In the steady-state GA the offspring are continually integrated with the population, thus offspring and parents occupy the same population pool every generation [77, 80].

The GA search process moves around in the search space using probabilistic rules rather than deterministic rules [77]. The probabilistic transition rules aid the algorithm to avoid local optima regions in the solution space [30].

Some sequential search algorithms require that the objective function be differentiable [62]. These sequential algorithms use the derivative to obtain gradient information so that they can move in the solution space [62,74]. The derivative of the objective function may be used in to increase the efficiency of the GA but is by no means a core requirement of the algorithm [30,62,74]. GA makes no assumptions about the solution space and primarily works on the information provided by the objective function [30, 62].

In this section we gave a overview of the Genetic Algorithm. We introduced the core concepts on which the algorithm is based upon as well as defined how the algorithm searches for solutions in a given domain. In the next section we will provide the pseudo code for the basic algorithmic structure as well as a flow diagram of the search procedure.

4.5.2 Important Genetic Algorithm characteristics

In this section we will give an overview on characteristics which make the GA search procedure unique. We will start of by discussing initial population generation, after which we will discuss the core operators that the GA uses. We will end of this section with a discussion on the efficiency of the GA.

Initial Population Generation

Initial population generation is the very first activity that the GA performs. Out of this population potential mating candidates are selected based on their fitness which indicates the desirability. Generally the initial population is generated by means of randomization [74]. Since the algorithm searches multiple points simultaneously in the solution space, it is desirable that the initial population have a wide diversity with regard to the solution they represent [17, 50]. By controlling the initial population generation we can control, to a small degree, the amount of exploration the algorithm does initially as well as avoid premature convergence [50]. Therefore, care must be taken in the selection of the particular randomization scheme that will be used to generate solutions.

In a survey done by Andrea Reese, two randomization schemes are defined namely pseudo-random number generators (PRNGs) and quasi-random number generators (QRNGs). PRNGs where found to be heavily problem dependant, improving the search efficiency in some instances and in other instances having no considerable impact. QRNGs on the other hand, were shown to significantly improve the final solution produced by the GA as well as lowering the number of generations for the solution to be obtained [61].

Not all GA algorithm use randomization entirely for their initial population generation. In research done by Amit Nagar, Sunderesh S. Heragu and Jorge Haddock an algorithm is presented that generates an initial population through some aid of a branch-and-bound algorithm. The branch-and-bound algorithm provides the GA with an upper bound of acceptable solution in the solution space. The initial population is then randomly generated in the constrained space defined by the upper bound [50].

Selection Operator

The Selection operator is the first operator to be applied to the population after each generation. This operator is in charge of evaluating the current population to determine which individuals will survive and which will be "killed" off [50, 62, 65]. The individuals who survive and thus have the highest fitness are moved to a "mating pool" [10]. Individuals from this mating pool will be used in the the reproduction phase to generate a new population [20, 32].

By favouring high fitness individuals above low fitness individuals the operator guides the search towards better high quality solutions [62]. But care must be taken if the operator is to keen on high quality individuals since it may eliminate diversity in the population and thus result in premature convergence for certain problems [62]. If the solution space is known to have only one optimum, then a strict selection policy may be used, therefore directing the search into a gradient based direction [62]. In contrast, with a solution space that is known to have multiple optima, a forgiving selection policy might be more favourable since it allows the solution space to be more widely explored [62].

The most widely adopted selection scheme is known as the *Roulette Wheel* selection scheme [62, 65, 80, 85]. With this scheme an individual is selected based on a probability defined by the fitness of the individual divided by the collective fitness of the population [80].

Crossover Operator

The Crossover operator is usually the first operator applied to the population in the reproduction phase. The crossover operates exclusively only on the chromosomes that are in the mating pool. This operator is the main process by which by the GA algorithm is able to diversify as well as exploit certain optimal regions [50,65]. Crossover works by interchanging and matching two parent chromosomes randomly selected from the mating pool to produce a single chromosome known as the offspring [10,65,77]. Since two chromosomes are combined or partially changed, some historical information is retained in the new chromosome [77].

In some algorithms like for instance the one presented in Nagar et. al., before the crossover operator is applied to the two parent chromosomes, the parents are first evaluated to determine if they represent suboptimal regions. If the either of the parents are from a suboptimal region, a disruption operator is applied that interchanges certain domain specific information between the parents. After the disruption operator is applied the crossover operator is applied [50].

There are a variety of ways with which values are interchanged between chromosomes in the crossover operation i.e, Fixed point crossover, Two Point Crossover, Uniform Crossover and Gaussian Crossover. Fixed point crossover operates on binary parents where by a point is selected in one parent and then all other bits are replaced by the other parents bits [10]. Two point crossover generates two random indices's which dictates a certain segment in the one parent to be interchanged with the other parent [62]. Uniform crossover is the most basic of all crossovers since it randomly selects bits from one parent to be replaced by another parents bits and is usually used when a large solution space must be search [78, 80]. Finally, the Gaussian crossover, interchanges bits between parents based on a Gaussian distribution [78,80]. Depending on the state of the algorithm, crossover operators can also be interchanged or even paired if the algorithm needs better search performance for large or small solution spaces [3, 78].

Note, in all above crossovers it is assumed that the chromosomes are bit encoded, but these crossover do require them to be. All these crossover operators are able to work on any encoding, it just depends on what is considered to be a "bit" if a non-binary encoding is used.

Mutation Operator

The mutation operator is a probabilistic operator, which means it is applied infrequently and thus only with a certain probability will it be applied to parent solution. The operator typically changes some small in the solution regardless of the fitness of the chromosome [10, 85].

Given enough time, the mutation operator enables the algorithm to search the entire search space [77]. The operator also aids the algorithm with regard to escaping local optima in the solution space [77].

Due to the way the crossover works, some information may be lost when it is replaced by another chromosomes bits [20, 62]. Mutation is a source of new information which is continuously inserted into the algorithm, hence it works against information loss [20, 62, 65]. Usually, the mutation operator has no previous information on the chromosome it is mutating, thus it is entirely possible that the mutation may modify the chromosome for the worse [20]. A worse solution might lead the algorithm out of local optima or lead it down a new path to find the global optima, but this isn't always the case and thus in the literature the probability of the mutation operator is set to be very low [32, 62, 77].

The mutation operator isn't always simple random operations. In research done by Il-kwon Jeong and Ju-jang Lee, a mutation operator is presented that incorporates the Simulated Annealing algorithm. Th SA mutation operator generates a new chromosome whose fitness is also calculated. If the new chromosome fitness is worse than the chromosome to be mutated, then Depending on the SA mutation temperature as well cooling schedule , the newly generated chromosome might replace the chromosome to be mutated. Otherwise, if the new chromosome has a better fitness than the chromosome to be mutated, then it is replaced [32].

Elitism Operator

The elitist operator differs from the crossover and mutation operator in a sense that, it doesn't modify the chromosomes in any way. Instead, the operator works on the population [57]. The elitist operator ensures that the best chromosomes do not get lost from one population to the next, when the crossover and mutation operators are applied [3]. Thus the elitist operator is only applied after the crossover and mutation operators have generated a new population.

The elitist selects a certain number of high quality chromosomes from the parent population and transfers them in the new population without any modification from the other two operators [57]. The operator does not just move parents into the new population. The operator typically replaces sub par chromosomes in the new population with the higher quality parents [27]. Therefore, the elitist operator helps the algorithm retain knowledge gained from previous generations and prevents the best solution from extinction [55]. Finally, the retainment of knowledge and best solution aids the algorithm with regard to global convergence [68].

Algorithm Efficiency

The GA is a powerful, yet simple algorithm and tends to find good solutions given enough time. But the algorithm does has its disadvantages. One of the major disadvantages occurs when the GA is applied to problems which have very large solution spaces. In these problem, the population size is a very sensitive parameter. If the population is too small the algorithm won't have enough diversity to search and tends to premature converge.

A large population is preferred in large problem spaces, but then the algorithm is very computationally expensive since more time is spent evaluating than evolving new populations and the speed of the algorithm convergence decreases drastically. Hence, the population size must be fine tuned to achieve optimal performance in large problem spaces

Another disadvantage of GA, is that it is memory intensive, most sequential algorithms search on a single point bases through the solution space. As discussed above(see page 50), searches multiple points simultaneously and therefore requires more memory the keep track of all the possible solutions.

Finally, the algorithm has some sort of hill-climbing through the mutation operator, but the probability of the mutation operator is far too low for the algorithm to be considered to have a real hill-climbing capability.

4.5.3 Algorithm and Data flow

In this section we will give the start of by providing the pseudo code of the Genetic Algorithm. We will also provide a flow diagram depicting the general search procedure flow of the algorithm.

4.6 Summary

Chapter 5

Swarm Intelligence

5.1 Introduction

Part II

Implementation

Bibliography

[1] Karen Aardal, Cor Hurkens, Jan Karel Lenstra, and Sergey Tiourine. Algorithm for radio link frequency assignment: The calma project. *Operations Research*, 50(6):968–980, 2002.

[2] Karen I. Aardal, Stan P. M. van Hoesel, Arie M. C. A. Koster, Carlo Mannino, and Antonio Sassano. Models and solution techniques for frequency assignment problems. *4OR: A Quarterly Journal of Operations Research*, 1(4):261–317, December 2004.

[3] E. Alba, F. Luna, A. J. Nebro, and J. M. Troya. Parallel heterogeneous genetic algorithms for continuous optimization. *Parallel Comput.*, 30(5-6):699–719, 2004.

[4] Mahmoud H. Alrefaei and Sigrn Andradttir. A simulated annealing algorithm with constant temperature for discrete stochastic optimization. *Management Science*, 45:748–764, 1999.

[5] Shawki Areibi and Anthony Vannelli. Circuit partitioning using a tabu search approach. In *In 1993 IEEE International Symposium on Circuits and Systems*, pages 1643–1646, 1993.

[6] A. Augugliaro, L. Dusonchet, and E. R. Sanseverino. An evolutionary parallel tabu search approach for distribution systems reinforcement planning. *Advanced Engineering Informatics*, 16(3):205 – 215, 2002.

[7] I. Badarudin, A.B.M. Sultan, M.N. Sulaiman, A. Mamat, and M.T.M. Mohamed. Metaheuristic approaches for optimizing agricultural land areas. In *Data Mining and Optimization, 2009. DMO '09. 2nd Conference on*, pages 28 –31, 27-28 2009.

[8] Ralf Borndrfer, Andreas Eisenbltter, Martin Grtschel, and Alexander Martin. The orientation model for frequency assignment problems, 1998.

[9] L. Cavique, C. Rego, and I. Themido. Subgraph ejection chains and tabu search for the crew scheduling problem. *The Journal of the Operational Research Society*, 50:608–616, 1999.

[10] A. Chawla, S. Mukherjee, and B. Karthikeyan. Characterization of human passive muscles for impact loads using genetic algorithm and inverse finite element methods. *Biomechanics and Modeling in Mechanobiology*, 8(1):67–76, 2009.

[11] Jeremy K. Chen, Gustavo de Veciana, and Theodore S. Rappaport. Site-specific knowledge and interference measurement for improving frequency allocatoins in wireless networks. *IEEE Transactions on Vehicular Technology*, 58:2366–2376, 2009.

[12] Agnieszka Debudaj-Grabysz and Zbigniew J. Czech. Theoretical and practical issues of parallel simulated annealing. In *PPAM'07: Proceedings of the 7th international conference on Parallel processing and applied mathematics*, pages 189–198, Berlin, Heidelberg, 2008. Springer-Verlag.

[13] Audrey Dupont, Andrea Carneiro Linhares, Christian Artigues, Dominique Feillet, Philippe, and Michel Vasquez. The dynamic frequency assignment problem. *European Journal of Operational Research*, 195:75–88, 2009.

[14] Andreas Eisenblätter. Assigning frequencies in gsm networks. Technical report, Konrad-Zuse-Zentrum für Informationstechnik Berlin (ZIB), 2001.

[15] Andreas Eisenblätter. *Frequency Assignment in GSM Networks: Models, Heuristics, and Lower Bounds.* PhD thesis, Technische Universität Berlin, Berlin, Germany, 2001.

[16] N. Fescioglu-Unver and M.M. Kokar. Application of self controlling software approach to reactive tabu search. In *Self-Adaptive and Self-Organizing Systems, 2008. SASO '08. Second IEEE International Conference on*, pages 297 –305, 20-24 2008.

[17] Gautam Garai and B. B. Chaudhuri. A distributed hierarchical genetic algorithm for efficient optimization and pattern matching. *Pattern Recogn.*, 40(1):212–228, 2007.

[18] Mohan Gopalakrishnan, Ke Ding, Jean-Marie Bourjolly, and Srimathy Mohan. A tabu-search heuristic for the capacitated lot-sizing problem with set-up carryover. *Manage. Sci.*, 47(6):851–863, 2001.

[19] J.S Graham, R. Montemanni, J. N J. Moon, and D. H. Smith. Frequency assignment. multiple interference and binary constraints. *Wireless Networking*, 14:449–464, 2008.

[20] Timo Hämäläinen, Harri Klapuri, Jukka Saarinen, Pekka Ojala, and Kimmo Kaski. Accelerating genetic algorithm computation in tree shaped parallel computer. *J. Syst. Archit.*, 42(1):19–36, 1996.

[21] Joe Rodriguez-Tellez Harilaos G. Standalidis, Peter P. Stavroulakis. An efficient evolutionary algorithm for channel resource management in cellular mobile systems. *IEEE Transactions of Evolutionary Computation*, 2(4):125–137, November 1998.

[22] Hassan M. Elragal Hassan M. Elkamchouchi. Channel assignment for cellular radio using particle swarm optimization. In *The 23rd National radio Science Conference (NRSC 2006)*, March 2006.

[23] Abdel-rahman Hedar and Masao Fukushima. Tabu search directed by direct search methods for nonlinear global optimization. *European Journal of Operational Research*, 170:329–349, 2006.

[24] Alan Herz, David Schindl, and Nicolas Zufferey. Lower bounding and tabu search procedures for the frequency assignment problem with polarization constraints. *4OR: A Quarterly Journal of Operations Research*, 3:139–161, 2005.

[25] L. Houssin, C. Artigues, and E. Corbel. Frequency allocation problem in a sdma satellite communication system. In *Computers Industrial Engineering, 2009. CIE 2009. International Conference on*, pages 1599 –1604, 6-9 2009.

[26] http://fap.zib.de/. Fap web, 2010.

[27] Shun-Fa Hwang and Rong-Song He. Improving real-parameter genetic algorithm with simulated annealing for engineering problems. *Adv. Eng. Softw.*, 37(6):406–418, 2006.

[28] Lhassane Idoumghar and Ren Schott. Two distributed algorithms for the frequency assignment problem in the field of radio broadcasting. *IEEE Transactions on Broadcasting*, 55:223–229, 2009.

[29] D.M. Jaeggi, G.T. Parks, T. Kipouros, and P.J. Clarkson. The development of a multi-objective tabu search algorithm for continuous optimisation problems. *European Journal of Operational Research*, 185(3):1192 – 1212, 2008.

[30] A.A. Javadi, R. Farmani, and T.P. Tan. A hybrid intelligent genetic algorithm. *Advanced Engineering Informatics*, 19(4):255 – 262, 2005. Computing in Civil Engineering.

[31] I. Jovanoski, I. Chorbev, D. Mihajlov, and I. Dimitrovski. Tabu search parameterization and implementation in a constraint programming library. In *EUROCON, 2007. The International Conference on Computer as a Tool*, pages 459 –464, 9-12 2007.

[32] Il kwon Jeong and Ju jang Lee. Adaptive simulated annealing genetic algorithm for system identification. *Engineering Applications of Artificial Intelligence*, 9(5):523 – 532, 1996.

[33] Sergio Ledesma, Miguel Torres, Donato Hernndez, Gabriel Avia, and Guadalupe Garca. Temperature cycling on simulated annealing for neural network learning. In *MICAI 2007: Advances in Artificial Intelligence*, pages 161–171, 2007.

[34] Yuming Liang and Lihong Xu. Mobile robot global path planning using hybrid modified simulated annealing optimization algorithm. In *GEC '09: Proceedings of the first ACM/SIGEVO Summit on Genetic and Evolutionary Computation*, pages 309–314, New York, NY, USA, 2009. ACM.

[35] Wen Liu, Haixiang Shi, and Lipo Wang. Minimizing interference in satellite communications using chaotic neural networks. *Natural Computation, 2007. ICNC 2007. Third International Conference on*, 2:441 –444, aug. 2007.

[36] Francisco Luna, Christian Blum, Enrique Alba, and Antonio J. Nebro. Aco vs eas for solving a real-world frequency assignment problem in gsm networks. In *GECCO '07: Proceedings of the 9th annual conference on Genetic and evolutionary computation*, pages 94–101, New York, NY, USA, 2007. ACM.

[37] H. Mahmoudzadeh and K. Eshghi. A metaheuristic approach to the graceful labeling problem of graphs. In *Swarm Intelligence Symposium, 2007. SIS 2007. IEEE*, pages 84 –91, 1-5 2007.

[38] S. Mallela and L.K. Grover. Clustering based simulated annealing for standard cell placement. In *Design Automation Conference, 1988. Proceedings., 25th ACM/IEEE*, pages 312 –317, 12-15 1988.

[39] Vittoria Maniezzo and Roberto Montemanni. An exact algorithm for the min-interference frequency assignment problem. Technical report, Department of Computer Science, University of Bologna, 2000.

[40] Asha Mehrotra. *GSM System Engineering.* Arteh House, Inc, 1997.

[41] David Meignan, Jean-Charles Créput, and Abderrafiaa Koukam. A co-operative and self-adaptive metaheuristic for the facility location problem. In *GECCO '09: Proceedings of the 11th Annual conference on Genetic and evolutionary computation*, pages 317–324, New York, NY, USA, 2009. ACM.

[42] George Jiri Mejtsky. The improved sweep metaheuristic for simulation optimization and application to job shop scheduling. In *WSC '08: Proceedings of the 40th Conference on Winter Simulation*, pages 731–739. Winter Simulation Conference, 2008.

[43] P.R.S. Mendonca and L.P. Caloba. New simulated annealing algorithms. In *Circuits and Systems, 1997. ISCAS '97., Proceedings of 1997 IEEE International Symposium on*, volume 3, pages 1668 –1671 vol.3, 9-12 1997.

[44] Marie-Bernadette Pautet Michel Mouly. *The GSM System for Mobile Communications.* Cell & Sys, 1992.

[45] Qi Ming-yao, Miao Li-xin, Zhang Le, and Xu Hua-yu. A new tabu search heuristic algorithm for the vehicle routing problem with time

windows. In *Management Science and Engineering, 2008. ICMSE 2008. 15th Annual Conference Proceedings., International Conference on*, pages 1648 –1653, 10-12 2008.

[46] Roberto Montemanni. *Upper and Lower bounds for the fixed spectrum frequency assignment problem.* PhD thesis, School of Tecnology, University of Glamorgan, 2001.

[47] Roberto Montemanni and Derek H. Smith. Heuristic manipulation, tabu search and frequency assignment. *Comput. Oper. Res.*, 37(3):543–551, 2008.

[48] Roberto Montemanni and Derek H. Smith. Heuristic manipulation, tabu search and frequency assignment. *Comput. Oper. Res.*, 37(3):543–551, 2010.

[49] Garry Mullet. *Wireless telecommunications systems and networks.* Thomsan Delmar Learning, 2006.

[50] Amit Nagar, Sunderesh S. Heragu, and Jorge Haddock. A combined branch-and-bound and genetic algorithm based approach for a flowshop scheduling problem. *Annals of Operations Research*, 63(3):397–414, June 1996.

[51] B. Natrajan and B.E. Rosen. Image enhancement using very fast simulated reannealing. In *Image Analysis and Interpretation, 1996., Proceedings of the IEEE Southwest Symposium on*, pages 230 –235, 8-9 1996.

[52] Lin Ni and Hong-Ying Zheng. An unsupervised intrusion detection method combined clustering with chaos simulated annealing. In *Machine Learning and Cybernetics, 2007 International Conference on*, volume 6, pages 3217 –3222, 19-22 2007.

[53] Koji Nonobe and Toshihide Ibaraki. A tabu search approach to the constraint satisfaction problem as a general problem solver. *European Journal of Operational Research*, 106(2-3):599 – 623, 1998.

[54] E. Nowicki and S. Zdrzalka. Single machine scheduling with major and minor setup times - a tabu search approach. *The Journal of the Operational Research Society*, 47:1054–1064, 1996.

[55] W. Paszkowicz. Properties of a genetic algorithm equipped with a dynamic penalty function. *Computational Materials Science*, 45(1):77 – 83, 2009. Selected papers from the E-MRS 2007 Fall Meeting Symposium G: Genetic Algorithms in Materials Science and Engineering - GAMS-2007.

[56] Thomas La Porta Patrick Traynor, Patrick McDaniel. *Security for Telecommunications Networks*, volume 40 of *Advances in Information Security*. Springer US, 2008.

[57] Pedro Pereira, Fernando Silva, and Nuno A. Fonseca. Biored - a genetic algorithm for pattern detection in biosequences. In *2nd International Workshop on Practical Applications of Computational Biology and Bioinformatics (IWPACBB 2008)*, pages 156–165, 2009.

[58] Jean-Yves Potvin. Genetic algorithms for the traveling salesman problem. *Annals of Operations Research*, 63:337–370, 1996.

[59] Nilesh B. Prajapati, Rupal R. Agravat, and Mosin I. Hasan. Comparative study of various cooling schedules for location area planning in cellular networks using simulated annealing. *Networks & Communications, International Conference on*, 0:146–150, 2009.

[60] Abraham P. Punnen and Y. P. Aneja. A tabu search algorithm for the resource-constrained assignment problem. *The Journal of the Operational Research Society*, 46:214–220, 1995.

[61] Andrea Reese. Random number generators in genetic algorithms for unconstrained and constrained optimization. *Nonlinear Analysis: Theory, Methods & Applications*, 71(12):e679 – e692, 2009.

[62] D. J. Reid. Genetic algorithms in constrained optimization. *Mathematical and Computer Modelling*, 23(5):87 – 111, 1996.

[63] Angelos N. Rouskas, Michael G. Kazantzakis, and Miltiades E. Anagnostou. Minimizing of frequency assignment span in cellular networks. *IEEE Transactions on Vehicular Technology*, 48:873–882, 1999.

[64] P. Demestichas E. Tzifa S. Kotrotsos, G. Kotsakis and V. Demesticha. Forumlation and computationally efficient algirithms for an

interference-orientated version of the frequency assignment problem. *Wireless Personal Communications*, 18:289–317, 2001.

[65] Mohsen Saemi and Morteza Ahmadi. Integration of genetic algorithm and a coactive neuro-fuzzy inference system for permeability prediction from well logs data. *Transport in Porous Media*, 71(3):273–288, February 2008.

[66] Patrick Siarry, Alain Pétrowski, and Mourad Bessaou. A multipopulation genetic algorithm aimed at multimodal optimization. *Adv. Eng. Softw.*, 33(4):207–213, 2002.

[67] J. R. Slagle, A. Bose, P. Busalacchi, and C. Wee. Enhanced simulated annealing for automatic reconfiguration of multiprocessors in space. In *IEA/AIE '89: Proceedings of the 2nd international conference on Industrial and engineering applications of artificial intelligence and expert systems*, pages 401–408, New York, NY, USA, 1989. ACM.

[68] K.G. Srinivasa, K.R. Venugopal, and L.M. Patnaik. A self-adaptive migration model genetic algorithm for data mining applications. *Information Sciences*, 177(20):4295 – 4313, 2007.

[69] Marc St-Hilaire, Steven Chamberland, and Samuel Pierre. A tabu search algorithm for the global planning problem of third generation mobile networks. *Comput. Electr. Eng.*, 34(6):470–487, 2008.

[70] Anand Prabhu Subramanian, Himanshu Gupta, Samir R. Das, and Jing Cao. Minimum interference channel assignment multiradio wireless mesh networks. *IEEE Transactions on Mobile Computing*, 7(7):1459–1473, December 2008.

[71] B. Suman and P. Kumar. A survey of simulated annealing as a tool for single and multiobjective optimization. *The Journal of the Operational Research Society*, 57(10):1143–1160, 2006.

[72] Ashish Sureka and Peter R. Wurman. Applying metaheuristic techniques to search the space of bidding strategies in combinatorial auctions. In *GECCO '05: Proceedings of the 2005 conference on Genetic and evolutionary computation*, pages 2097–2103, New York, NY, USA, 2005. ACM.

[73] Ashish Sureka and Peter R. Wurman. Using tabu best-response search to find pure strategy nash equilibria in normal form games. In *AAMAS '05: Proceedings of the fourth international joint conference on Autonomous agents and multiagent systems*, pages 1023–1029, New York, NY, USA, 2005. ACM.

[74] Vedat Toğan and Ayşe T. Daloğlu. An improved genetic algorithm with initial population strategy and self-adaptive member grouping. *Comput. Struct.*, 86(11-12):1204–1218, 2008.

[75] Nguyen Thanh Trung and Duong Tuan Anh. Comparing three improved variants of simulated annealing for optimizing dorm room assignments. In *Computing and Communication Technologies, 2009. RIVF '09. International Conference on*, pages 1 –5, 13-17 2009.

[76] Hsien-Yu Tseng and Chang-Ching Lin. A simulated annealing approach for curve fitting in automated manufacturing systems. *Journal of Manufacturing Technology Management*, 18:202 – 216, 2007.

[77] Roger L. Wainwright. A family of genetic algorithm packages on a workstation for solving combinatorial optimization problems. *SIGICE Bull.*, 19(3):30–36, 1994.

[78] Z. G. Wang, Y. S. Wong, and M. Rahman. Development of a parallel optimization method based on genetic simulated annealing algorithm. *Parallel Comput.*, 31(8+9):839–857, 2005.

[79] N. A. Wassan. A reactive tabu search for the vehicle routing problem. *The Journal of the Operation Research Society*, 57:111–116, 2006.

[80] Xian-Huan Wen, Tina Yu, and Seong Lee. Coupling sequential-self calibration and genetic algorithms to integrate production data in geostatistical reservoir modeling. In *Geostatistics Banff 2004*, volume 14 of *Quantitative Geology and Geostatistics*, pages 691–701. Springer Netherlands, 2005.

[81] Dennis Weyland. Simulated annealing, its parameter settings and the longest common subsequence problem. In *GECCO '08: Proceedings of the 10th annual conference on Genetic and evolutionary computation*, pages 803–810, New York, NY, USA, 2008. ACM.

[82] Lihua Wu and Yuyun Wang. An introduction to simulated annealing algorithms for the computation of economic equilibrium. *Computational Economics*, 12:151–169, 1998.

[83] Yiliang Xu, Meng Hiot Lim, and Yew-Soon Ong. Automatic configuration of metaheuristic algorithms for complex combinatorial optimization problems. In *Evolutionary Computation, 2008. CEC 2008. (IEEE World Congress on Computational Intelligence). IEEE Congress on*, pages 2380 –2387, 1-6 2008.

[84] Dominic C O'Brien Yangyang Zhang. Fixed channel assignment in cellular radio networks using particle swarm optimization. In *Proceedings of the Internation Symposium of Industrial Electronics*, June 2005.

[85] Quan Yuan, Feng Qian, and Wenli Du. A hybrid genetic algorithm with the baldwin effect. *Information Sciences*, 180(5):640 – 652, 2010.

[86] L. Zhang, S. Guo, Y. Zhu, and A. Lim. A tabu search algorithm for the safe transportation of hazardous materials. In *SAC '05: Proceedings of the 2005 ACM symposium on Applied computing*, pages 940–946, New York, NY, USA, 2005. ACM.