2012-01-01

Implementation Of Evolutionary Algorithms On The Power Industry And Aviation Security

Anuar Jesus Aguirre
University of Texas at El Paso, ajaguirre3@miners.utep.edu

IMPLEMENTATION OF EVOLUTIONARY ALGORITHMS ON THE POWER INDUSTRY AND AVIATION SECURITY

ANUAR JESUS AGUIRRE NUÑEZ

Department of Industrial Manufacturing and Systems Engineering

APPROVED:

Jose F. Espiritu, Ph.D., Chair

Heidi A. Taboada, Ph.D.

Bill Tseng, Ph.D.

Valentin Bolborici, Ph.D.

Benjamin C. Flores, Ph.D.
Interim Dean of the Graduate School

Dedication

To my brother and my parents

For all their sacrifices, support and unconditional love throughout my life

IMPLEMENTATION OF EVOLUTIONARY ALGORITHMS IN THE POWER INDUSTRY AND AVIATION SECURITY

by

ANUAR JESUS AGUIRRE NUÑEZ,

Bachelor of Science in Industrial Engineering

THESIS

Presented to the Faculty of the Graduate School of

The University of Texas at El Paso

in Partial Fulfillment

of the Requirements

for the Degree of

MASTER OF SCIENCE

Department of Industrial Manufacturing and System Engineering

THE UNIVERSITY OF TEXAS AT EL PASO

May 2012

Acknowledgements

First and foremost I offer my sincerest gratitude to my supervisor, Dr Jose Espiritu, who has supported me throughout my thesis and with my personal life during the past two years. I attribute the level of my Masters degree to his encouragement and effort and without him this thesis, too, would not have been completed or written. Also I want to thank Dr Heidi Taboda, for teaching me all I know about metaheuristics, for always motivate me and for her guidance during the last two years.

A special thanks for all my friends Karla Gutierrez, Olivia Moreno, Abril Vazquez, Perla Torres, Carlos Ituarte, Oswaldo Aguirre, Aaron Martinez, Nicolas Lopez and Edgar Jimenez for helping whenever I have I doubt in my classes and during my research, but mostly because giving me a friendship for all my life.

Finally, I want give my whole gratitude to my parents for always understand me, believe in me and for showing me that life is not easy and we need to fight for whatever I want. An special thank to my deceased brother who during his life always support me in my decisions and for inspire me to always try do my best even when everything looks bad, I know that he is still with me.

Table of Contents

List of Tables

List of Figures

Chapter 1: Introduction

1.1 Research Background

The present thesis is concerned in the use of metaheuristics to find excellent solutions for hard problems that our society is facing. A persistent rise in the importance of the optimization on the use of resources is one of the major trends of the human history. Since the industrial revolution that began in 1970s the industries began to try to use the resources in a more intelligent way. Industries and business always try get as most as possible outcome from what they are producing and/selling. In a remarkable profusion of applications, optimization has penetrated deeply into every area of human life, for example, industries, hospitals, home, and also in different growing commercial and service sectors making important changes on to them. Optimization is undoubtedly one most integral part of human contemporary life and a key part on human development in various sectors. The lack of optimization techniques causes a economical lost and a various steps behind of a comparable industry that uses it, due to the reduced commercial and industrial production. The primary function of an optimization technique is to create a reliable method to catch the extreme of a specific function by an intelligent arrangement of its evaluations (measurements).

Our society is highly dependent on time and cost-effective solutions, and two of the most important actual problems are the security on airports and the reliability on power systems, since air transportation and electricity are essential services. Therefore, having a reliable power system and an excellent airport security are very important factor for a good life quality. In order to provide constant electricity supply we need to design a component replacement schedule for the electric distribution systems. One of the key aspects in designing the component replacement schedule is the reliability. An unreliable power system infers in high cost to the electric company and most important to the customers.

According to SGI Federal (2003), the cost of a major power outage confined to one state can be on the order of tens of millions of dollars per day. One power outage that affects multiple states can cost over one hundred million dollars. A power grid consists in three main divisions, which are: the generating station, transmission network, and distribution network. The generation station is where the electricity is produced; the transmission network is the connection between the station where the electricity is generated and the distribution network. A bulk supply point is the point of connection between a distribution and a transmission network. The distribution network links a utility bulk transmission network and the retail customer.

For power systems, the number and duration of supply interruptions characterize the continuity of supply (Sand *et al.,* 2004). A reliable power system is a system that has the ability of supply electricity at any point of time. Sometimes in the electrical industry the term "availability" is used instead as "reliability". One of the main focuses of this research is in the power distribution system.

For the power industry most of the time the distribution system begins at the substation and ends with the final customer. Most of the customer electrical outages (up to 80%) occur due to a failure in the distribution system. Since the demand of electricity is increasing constantly it is basic for the power industry to have a very reliable electric distribution system. A component replacement analysis is the determination of when to change a specific component in the system, in order to minimize the impact of a failing component. The impact of a failure component can be determined in cost such as the cost of an outage. At each planning period two decisions should be taken: one is to either KEEP the component or REPLACE it for a completely new component. When a component is selected to continue is the system (keep it) the component must receive maintenance in order to continue its operations, as it wears out with the aging process.

In the power system we can divide the component replacement problem in two different categories: series replacement or parallel replacement. In a series replacement problem one or multiple

independent components, that interrupts the service when at least one fails, are considered to be replaced at a specific time. On the other hand, in a parallel replacement problem multiple interdependent components, that interrupts the service when a specific combination of components fail, are considered to be replaced at a specific time. When some constraints are added to the replacement problems (either series or parallel), the solutions search (KEEP or REPLACE) results in a complex combinatorial optimization problem.

The other important issue that this thesis addresses is the aviation security problem. For the threats of terrorism against aircraft during the past years has received more attention due to the security concerns. For this reason airport screening procedures all over the world and more in the United States have gone through important improvements in order to ensure safety. In 1996 the White House Commission on Aviation Safety and Security (CASS) was created due to the crash of the Trans World Airlines Flight 80 (TWA 800). The mayor contribution of the CASS was the recommendation of the deployment and use of new screening technologies and equipment, also the development of standards for training and testing the screening security device. In 1996 the United States congress passed the Federal Aviation Reauthorization Act of 1996. By the same time in 1997 the United States Congress passed the Omnibus Consolidated Appropriations Act. Finally in 2000, the United States Congress provided to the Federal Aviation Administration (FAA) a found of one billion dollars of airport security. One third of this founding was used to purchase and deployment of security equipment (Coughlin *et al.*, 2002). One of the major improvements in the airport security was reach by the use of Explosive Detection Systems, the implementation of Passenger-Baggage Matching and Automated Passenger Profiling. The Transportation Security Administration (TSA) has mainly focused in identifying potential threats by baggage screener training and the implementation of new procedures.

By finish of the 20th century, a system was implemented to improve the airport security by the identification of potential terrorists through the use profiles; this system is the Computer-Aided

Passenger Prescreening System (CAPPS). CAPPS helps the security personnel to focus their attention on the higher risk individuals. When a passenger is classified as not a risk a label of *nonselectee* is used over it, while in the other hand those found as a possible threat is classified as *selectee* (O'Harrow, 2002). According to previous research the implementation of CAPPS is not sufficient to warranty the total security and also it was found that some airports using CAPPS in order to select passengers for a more intense security process are less secure than other airports that use a random selections of passengers that go through inspection. For example, in 2002 Chakrabarti and Strauss developed Carnival Booth, an algorithm that uses a combination of statistical analysis and computer simulation, to demonstrate how a terrorist could defeat CAPPS. Chakrabarti and Strauss evaluated the efficacy of their algorithm and demonstrate that CAPPS is an inefficient method for airport security.

The breakpoint in the airport security was the terrorist attacks on September 11, 2001. Since 2001 aviation security changed to a uniform screening across the country with the law that mandates 100% checked baggage screening, eliminating the distinction between passengers' *selectees* and *nonselestees*. Even though, the TSA revisited the selective screening policies by the development of CAPPS II, a continuation of CAPPS, but on July 14, 2004, the TSA announced that CAPPS II would not be implemented. But, TSA announced that by the research done to implement CAPPS II, it was found that it is more effective to perform a more intense scrutiny of passenger that has a profile of "security risk" than increasing the security for all passengers. Poole and Passantino in 2003 said that is not cost-effective to implement a 100% checked baggage screening policy and proposed to create multiple levels of security for screening passengers.

CAPPS II only improve aviation security under some particular set of circumstances, so that it is recommended that CAPPS II be transitioned from security centerpiece to one part of many future components in aviation security (Barnett, 2004). In 2006 Martonosi and Barnett created a mathematical model that explores the effectiveness of airport passenger prescreening system against terrorist attracts

and noticed that CAPPS II may not improve the security of aviation. MITRE Corporation and Weiss developed in 2011 the Dynamic Airport Security Model. This model is a fast-time desktop simulation that accepts the airport layout, security procedure and threat vectors (path-weapon combinations) as the main inputs and with that information develop a model of the performance of the airport's security. Most of the airports today use many levels of security specially to check the baggage, using in most of them Explosive Detection Systems. There are many different types of Explosive Detection Devices that has different output. When an airport needs to find the best combination of levels and machines to use also results in a complex combinatorial optimization problem.

Since both problems previous described are combinatorial optimization problems this research focuses in the implementation of metaheuristics to solve those important issues that our society is facing. One of the most intense growing research area is the use of metaheuristics to solve combinatorial optimization problems. One of the main reasons for this is that the combinatorial optimization problems are very important for the scientific world, but are even more important for the industry. All the optimization problems try to find the best combination of the given variables to achieve an specific goal. For the combinatorial optimization problems the main goal is to find a discrete mathematical object minimizes/maximizes the objective function. The present research uses Genetic Algorithms to solve the component replacement schedule in power distribution systems problem and Memetic Algorithms to solve the design of an optimal baggage screening strategy.

Memetic Algorithms (MA's) and Genetic Algorithms (GAs) are a class of search and optimization methods inspires by the evolutionary adaptation in nature. GA's were introduced in early 1970s by Holland, but were implemented in optimization problems until late 1980s by Goldberg. While the combination of an Evolutionary Algorithm (EAs) with a local search makes a "Memetic Algorithm" (Moscato, 1989). In both cases simple GAs and MAs are global optimization methods, since the first

trial solutions run by them are based on global information that is then utilized by the search process. In both cases the GAs and MAs the optimization mechanism are the following:

- GAs and MAs uses a population of chromosomes, where each chromosome represents a trial solution of the problem.
- For each chromosome its fitness value its evaluated, where the fitness value is the criterion of the optimization problem evaluated.
- For each generation, the chromosomes that gave the fitness value are selected to undergo through a series of evolutionary operators to produce new chromosomes (sometimes called offspring).
- The better chromosomes then evolve by the optimization process, and by then the fittest chromosome is the optimized solution.

As motioned before the solutions of the optimization problem are codified chromosomes (Krasnogor and Smith, 2005). GAs and MAs uses evolution operators, and in for them they are the same two evolution operators, the crossover operator and the mutation operator. GAs and MAs begin to work with a given population (set) of random solutions in form of chromosomes, where the fitness value of each chromosome is determined by the evaluation of the objective function (Kamepalli, 2001). The evolutionary operators help the best chromosomes to interchange information to produce new chromosomes. After the evolutionary operator process the new set of solutions are then evaluated and used to make a continuous evolution. The process is repeated for a specific number of generations to obtain a optimum solutions, or at least a near to optimum solution to the combinatorial optimization problem.

Sometimes, classical GAs is not aggressive enough to solve some combinatorial optimization problems and should be enhanced with local search methods (Mendoza *et al;* 2009). The difference between a GA and a MA is the use of a local search operator extra for the MA. A local search helps the MA to locate a local optimum at each iteration more efficiently than GAs (Garg, 2009). The term

'Memetic Algorithms' was adopted in the late 80's in order to denote a new family of metaheuristics that combines tightly separated families such as evolutionary algorithms (Moscato *et al;* 2004).

1.2 Research Objectives

The present work shows how evolutionary algorithms can be used in order to solve hard combinatorial problems. Two different industries with highly different demands were chosen to demonstrate the functionality and flexibility of evolutionary algorithms to solve combinatorial problems. For the component replacement scheduling model for the power distribution system over finite planning horizon a Genetic Algorithm was used. A model is developed for a radial configuration and for a complex configuration. Radial configuration is the most commonly used in the power industry but that tendency is moving towards a complex configuration. The main objective of the research focus in the power industry is to find the replacement schedule that minimizes the total cost, subject to budget constraints.

In order to minimize the total cost for the component replacement schedule, another objectives should be addressed. For example, first of all a generalized formulation for the main goal in which all the issues are examined should be done. Most of the issues that should be addressed are the objectives and constraints that are commonly encountered in real life.

In the other hand using a Memetic Algorithm solves the second problem addressed by this research. A model was developed for different number of levels, from two up to ten. For space constraints ten is the maximum number of levels allowed for a common airport. The main objective of the second half of the present research is to find the best configuration of machines and levels that minimizes the total cost.

1.3 Proposed Thesis Layout

In chapter 2 metaheuristics approaches and exact optimization methods were presented. In the first section of the chapter describes what is a combinatorial optimization problem. The second section of the chapter gives a brief explanation of some exact optimization problems. The exact optimization methods explained are: Linear Programming, Integer Programming, and Dynamic Programming. The third section of the chapter explains what a metaheuristic is and how it works. Different types of metaheuristics are explained such as Tabu Search and different evolutionary algorithms. The evolutionary algorithms explained are: Ant Colony Optimization, Particle Swarm Optimization, Monkey Algorithm, Genetic Algorithm and Memetic Algorithm. Since the Genetic Algorithm and Memetic Algorithm are the evolutionary algorithm used in the present research one section of the chapter is completely dedicated to them.

Chapter 3 gives a main description of the component replacement schedule for the power distribution system and how the power system works. In the first section it is explained what is component replacement analysis. The second section explains the concepts of electricity generation, electricity transmission and the distribution systems. Each stage of the power system network is explained in detail. The third section of the chapter is focused in explain the distribution system, since it is the main topic of the first half of the research. The third section explains how the distribution network works, which factors and components are involved, and finally the main importance of the distribution system.

Chapter 4 explains the model developed to solve the component replacement scheduling. Chapter 4 gives a detail explanation of the methodology and formulation necessary to solve the problem. Section 2 gives a literature review of the application of similar metaheuristic to find an optimal solution in different problem domains. Section three presents the application of Genetic Algorithms in different component replacement problems. Section four present the Non-Homogenous Poisson Process (NHPP)

method. NHPP is used to calculate the aging process of the components of the distribution system. Section four also present the different formulation sued to calculate the total cost of schedule. The total cost is the summation of the maintenance cost, the unavailability cost and the purchase cost. Also, presents the Genetic Algorithm developed. Two configurations were solved for the component replacement analysis. A algorithm is developed for the radial configuration and the complex configuration. Section 1 present the algorithm developed for the radial configuration, with a detailed explanation of the steps involved to find the optimal solution. The second section explains the difference between the formulations to find solve the radial and the complex configurations. The third section of the chapter gives an example of the complex configuration and the best schedule obtained for it.

Chapter 5 gives a main description of the aviation baggage-screening problem. The first section gives an introduction to the baggage-screening problem along with a literature review. The second section explains the basic concepts of the baggage-screening problem.

Chapter 6 explains the methodology to solve the problem and the model developed. In section one the problem is analyzed and shows the mathematical model of the problem. The second section describes the model assumptions. Section three explains the Memetic Algorithm and Genetic Algorithm used. Section four gives an example of the baggage-screening problem and explains the results.

Chapter 2: Heuristic Optimization Methods

2.1 Introduction

The main focus of this chapter is to introduce and understand the different methods of optimization techniques. The different optimization methods are explained briefly with an special emphasis in the explanation of the Genetic Algorithms and Memetic Algorithms. First a brief description of what a combinatorial optimization problem is presented. Afterwards the exact optimization methods are described, with special emphasis in Linear Programming, Integer Programming and Dynamic Programing. Then, the metaheuristic method is explained. Tabu Search, Ant Colony Optimization, Particle Swarm Optimization, Monkey Algorithm, Genetic Algorithm and Memetic Algorithm are explained.

2.2 Combinatorial Optimization Problems

When a best solution is found over a set of feasible solution we say that the problem solved was optimized. All problems that could be optimized try to find the best combination of values that minimize or maximize the objective function. The generalized optimization theory covers a large area of mathematics. Optimizing is to find "best value" of the objective function.

In recent years the term "Combinatorial Optimization" (CO) has emerged. CO is used to describe the areas of mathematical programming concerned to the solution of optimization problems that has a combinatorial structure. Humanity has faced optimization problems since ancient ages; the main problem with optimization problems is that sometimes they have an infinite number of solutions. In recent years it has been found that most of the recent optimization problems have a finite number of solutions, even though the set of solutions is pretty big. Combinatorial problems are focused on the techniques and theory of the problems that have a finite number of solutions and the main goal is to find the best solution, where the set of feasible solutions is discrete or can be reduced to discrete. Software

engineering, mathematics and Artificial intelligence are some important applications of the operations research and computational complexity theory related to combinatorial optimization. Most of the real world problems such as scheduling, assignment, routing, packing, cutting, network design are combinatorial optimization problems. The most common combinatorial optimization problems are:

- Traveling Salesman Problem: For a set of different cities, which will be the best path to follow if all the cities must be visited and must return to the start point. The main objective is to minimize total distance traveled.
- Facilities Layout Problem: A set of facilities that need to be laid on the space of a factory. The main objective is to maximize the benefit of the location.
- Vehicle Scheduling Problem: A set of cars that need to visit a number of locations. The main objective is to minimize the distance traveled, with the constraint of vehicle capacity.
- Transportation Problem: For a set of warehouses to a set of factories design a distribution system. The main objective is to minimize the transportation cost for each factory and warehouse demand.

The last examples are the most commonly hard combinatorial optimization problems in real world, but they are not the only and most important real-world combinatorial optimization problems. There are two different types of techniques to solve combinatorial optimization problems. One technique is the exact optimization methods such as, Linear Programming, Integer Programming and Dynamic Programming. The second category of techniques is the metaheuristic optimization such as Genetic, Monkey and Memetic Algorithms. Other examples of metaheuristics are the Particle Swarm Optimization, Tabu Search and Ant Colony Optimization.

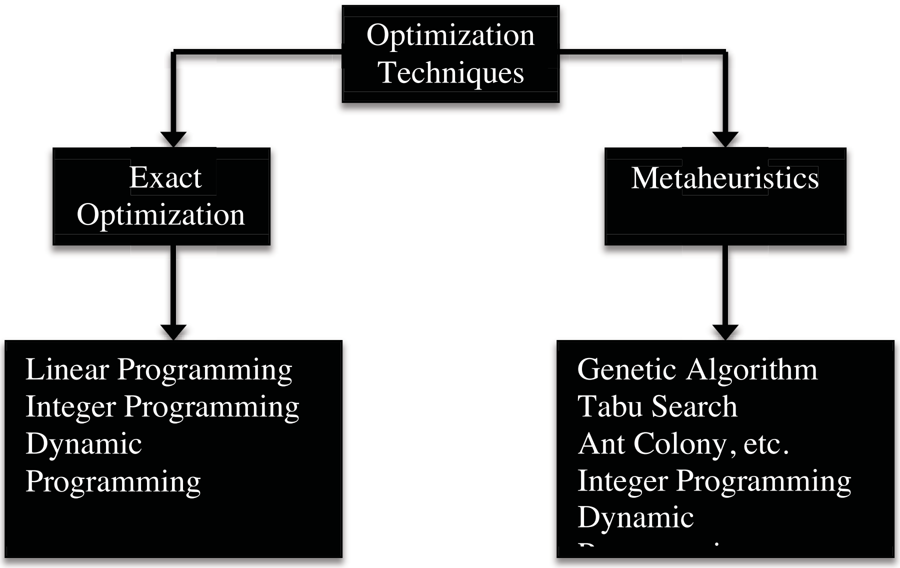


Figure 2-1: Optimization Techniques to solve Combinatorial Problems

2.3 Exact Optimization

The main advantage of the exact optimization methods is that they do guarantee to find the best solution over the set of feasible solutions. The main disadvantage is the run-time to find the optimal solution. The run-time for the exact optimization methods increases exponentially as the size of the problem increases. Most of the time only small optimization problems can be solved to prove the optimality of the solution found.

2.3.1 Linear Programming

Linear programming problem is a problem that has linear constraints and a linear function subject to the constraints that needs to be either maximized or minimized. In operations research the most used optimization technique is Linear Programming (Zoints, 1974). A constraint is a formula that limits the feasibility of a solution; the constraints must be represented as inequalities and equalities. LP takes all the constraints related to the same situation and finds the BEST combination for the main objective satisfying the given constraints. To solve a linear programming problem the objective function

and the constraints must be linear. A example of a linear programming problem is the problem that a farmer faces when has a limited amount of money to buy seeds to plant and has only 10 acres to use, he wants to maximize its profit but each different seed would need a different amount of space and will give a different profit, so which is the best combination of seed to maximize profit. Linear Programming is used everyday for the organization and allocation of resources. Since linear programming is extensively used in economics, linear programming is one of the most important optimization techniques.

Simple Linear Programming problems are solved graphically by plotting the inequalities/equalities formulas that represent the constraints of the problem. The constraints forma a bounded are in a x-y plane. The area bounded is called the feasible region, and the corners of the feasible region are the candidates of the optimal solution. To find the value of the corners a pair of lines that intersects must be solved in point of the intersection. Once the coordinates of the intersection points were found those values are tested in the optimization equation, in which we are trying to find to minimum or maximum value. Figure 2 shows the graphic representation of the linear programming method, where the area in blue is the feasible region.

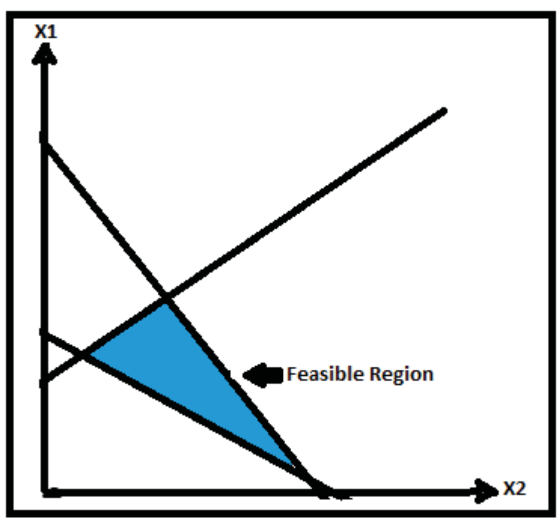


Figure 2-2: Graph of the bounded region of a Linear Programming problem

Only simple linear programming problems can be solved using the graphic method. When the number of variables increases and the problem becomes more complex the graphical method does not solve the linear programming problem. If the linear programming problem is complex the simplex method and dual simplex method are the most widely used technique. The standard form of a linear programming problem is shown below:

Maximize: **CX** (linear function)

Subject to,

AX=b (constraints)

Figure 2-3: Standard form of Linear Programming Problem

2.3.2 Dynamic Programming

Dynamic Programming solves a problem by identifying a set of sub problems. Each sub problem is solved one by one, from lower to higher. The smallest problem is solved first and the answer of the last problem will help to solve the next in size problem until the main problem is solved. Here the longest common subsequence is ABAD. Dynamic Programming tries to simplify highly complicated problems by breaking them into sub problems in a recursive manner. In 1955 Bellman introduced the Dynamic Programming formulation to solve a finite horizon equipment replacement. This problem addressed general costs and considers a single challenger in each decision period.

For Bellman the "principle of optimality" is that the some decision problems cannot be divided in sub problems, then the decisions span several points in time often break apart. If the sub problems can nested into larger problems, the dynamic programming methods can be used. Bellman also shown that can be stated in a step-by-step form. This can be achieve by writing down the relationship between the value function in the actual period with the value function and the next period, this is called the Bellman

equation. In order to find a optimal solution using dynamic programming the Bellman equation is necessary. The Bellman equation gives the utility of a state, when there are “n” states there are “n” Bellman equations. This gives a system of simultaneous equations, but in this case the equations are not linear because of the max/min operator.

In recent years Espiritu *et al;* (2008) have used a combination of Dynamic and Integer Programming to obtain a cost-efficient system level component replacement schedule. The model solves capital replacement problems for a set of heterogeneous assets in a power system grid subject to budget constraints. The main objective of the research was to minimize the total net present value of unmet demand. The model considers the system availability, maintenance, and purchase cost over a finite planning horizon. For maintenance scheduling problems dynamic programming is one of the best options, mainly for its properties (Yamayee *et al;* 1983).

2.3.3 Integer Programming

When in a optimization problem the variables are restricted to be integers the best option to solve the problem is Integer Programming. In some special cases not all the variables are required to be an integer, in those cases the problem is a mixed integer-programming problem. For the combinatorial problem of integer programming the feasible region is not a convex set, making it more difficult than a simple linear programming problem. In integer programming the problems have many local optima, such that finding the global optimum requires to prove that the solution dominates all the other feasible solutions. The standard form of Integer Programming Problem is shown in figure 2-4.

$$\text{Maximize: } \boldsymbol{f}(\mathbf{X})$$

Subject to,

$$\boldsymbol{g_j}\,(\mathbf{X}) = \mathbf{0}, \qquad j = 1, 2, \ldots, m,$$

$$\boldsymbol{h_i}\,(\mathbf{X}) \leq \mathbf{0}, \qquad i = 1, 2, \ldots, k,$$

$$\mathbf{X} = (x_1, x_2, \ldots, x_q, x_{q+1}, \ldots, x_n),$$

where, $x_1, x_2, \ldots, x_q$ are integers for a given q.

Figure 2-4:Sandard form Integer Programming Problem

2.4 METAHEURISTICS

Algorithm is a scientific procedure, which converge to the best solution of a given problem, in the other hand an heuristic is an algorithm that most of the time gives a solution close to the right answer (Black, 2004). Most of the real-world problems are so complex that standard algorithms do not have the capacity solve them. Complex combinatorial problems are so large that the computational time to solve them using exact optimization techniques can be solve, one example of this kind problems are the vehicle routing problems and the traveling sales problem. Also a complex combinatorial problem is when the problem cannot be formulated in explicit terms, neither in quantitative terms.

One of the highest areas of research is the possibility of applying metaheuristics due to its nature to find close to optimal solutions, and in some cases the optimal solution. There is no any precise definition of metaheuristic, one widely accepted definition is the one gave by Goldberg in 1989, “A metaheuristic refers to a master strategy that guides and modifies other heuristics to produce solutions beyond those that are normally generated in a quest for local optimality.” Another accepted definition for a metaheuristic is “A metaheuristic is a general algorithmic framework which can be applied to different optimization problems with relatively few modifications to make them adapted to a specific problem” presented Dorigo in 1996. The most recent definition of metaheuristic was gave by Black in

2009: “Metaheuristic can also be defined as a high-level algorithmic framework or approach that can be specialized to solve optimization problems.”

All the metaheuristics are strategies that guide a problem specific heuristic to boost its performance. When using a metaheuristic is important to focus in avoid to be trapped in a local minima/maxima. To not be trapped at a local minima/maxima, the metaheuristic uses different approaches. For example, in way to move from a local minima is to generate new starting solution for the heuristic used. The metaheuristic optimization methods use probabilistic decisions during the search for solutions. The main difference between a random search and a metaheuristic, is that the metaheuristic used the randomization is not completely blind, but instead intelligent, in a biased form (Stützle, 1999).

Metaheuristics are categorized in different ways depending on its characteristics. There is nature inspired metaheuristics and non-nature inspired metaheuristics. Also there is memory-less metaheuristics and memory usage metaheuristics, which are differentiated by the use of short-term memories or long-term memories. Another categorization is the single point and population based metaheuristics (Blum *et al;* 2008).

2.4.1 Particle Swarm Optimization

Particle Swarm Optimization (PSO) is a metaheuristic inspired in the social behavior of organism, more concrete in birds and fishes. Particle Swarm Optimization mimics the collective behavior of the birds’ crowds. James Kennedy and Russell Eberhart introduced particle Swarm Optimization in 1995. Particle Swarm Optimization is a population based metaheuristic that is inspired in the mutual cooperation. Particle Swarm Optimization is also a tool used in socio-cognition of human and artificial causes, and is based on the principles of social psychology (Kennedy *et al;* 1995). Particle Swarm Optimization has the following five principles:

1 The population carries space and time computations (proximity principle).

2 The population responds to the factors in the environment (quality principle).

3 The population change in behavior when it is needed (adaptability principle).

4 The population does not change its behavior every time the environment changes (stability principle).

5 The population should not commit its activities along excessively narrow channels (diverse response principle).

For Particle Swarm Optimization each solution is called "bird" in the "particle" search space. The birds have a fitness value that is evaluated by the fitness function. The particles also have velocities that direct the flying of the bird, and then the birds fly in the problem space following the local optimum solutions. Particle swarm optimization starts with a group of random birds (solutions) and it searches for the optimal solution by updating the generations of "birds". At each iteration every bird is changed by the following two "best" values. At the end the first bird is said to be the best solution found. The best solution found is then called *pbest*. When all the swarms find the "best" value is called *gbest* it is a global optima. The local best value found is then called *lbest*.

On each dimension the bids velocities are limited to a maximum velocity *Vmax*. *Vmax* helps to limit the velocity of the sum of the accelerations to do not exceed the maximum velocity defined by the programmer.

In other words the Particle Swarm Optimization works as follows: For each bird its position and velocity vectors are random selected, then it measures the fitness of each bird (*pbest*) and then the bird is stored with the best fitness (*gbest*) value. After that the velocity and positions vectors according to first two steps for each bird. Finally the last two steps are repeated until the stopping criterion is satisfied. The main advantage of the Particle Swarm Optimization is that it has few parameters to adjust, and it is widely used in multi-objective optimization, biological system modeling, pattern recognition, system

design, image segmentation, job shop scheduling and scheduling and planning. The pseudo code of the procedure is as follows

```
For each particle
   Initialize particle
END

Do
   For each particle
      Calculate fitness value
      If the fitness value is better than the best fitness value (pBest) in history
         set current value as the new pBest
   End

   Choose the particle with the best fitness value of all the particles as the
gBest
   For each particle
      Calculate particle velocity according equation (a)
      Update particle position according equation (b)
   End
```

Figure 2-5: Particle Swarm Optimization pseudo code

2.4.2 Tabu Search

Tabu Search is a procedure used in other metaheuristics in order to avoid be trapped at local optimal solutions. In 1986 Glover proposed Tabu Search as a metaheuristic to solve various combinatorial problems, especially problems addressed by the operations research literature. Most of the time Tabu Search finds solutions very close to the optimal solutions, sometimes it finds the optimal solution. Using Tabu Search we can tackle difficult problems at hand, making it one of the most popular metaheuristics to solve combinatorial problems at practical settings. Tabu Search was primarily

designed to find solutions for hard combinatorial problems; most of them are the Traveling Salesman Problem, routing problem, and Job Shop Scheduling.

Tabu Search has the distinguishing feature of adaptive forms of memory, which equips it to penetrate complexities that often confound alternative approaches (Glover *et al;* 1997). The most important component of Tabu Search is the Tabu List, so that in order to begin a Tabu List needs to be created. Then an initial solution *(s)* is set at random to be alter by a local search to scan neighborhoods, *N(s)*. From all the neighbors *(k)* of the solution *s,* the best neighbor is selected *(s')*. The combination that gives the best solution goes to the Tabu list. Then, at each iteration the best neighbor found is going to be the solutions that at the next iteration will be alter to find new neighbors. This process is repeated until the stopping criterion is met. The best solutions found at the end of the Tabu Search process is said to be the "optimal solution".

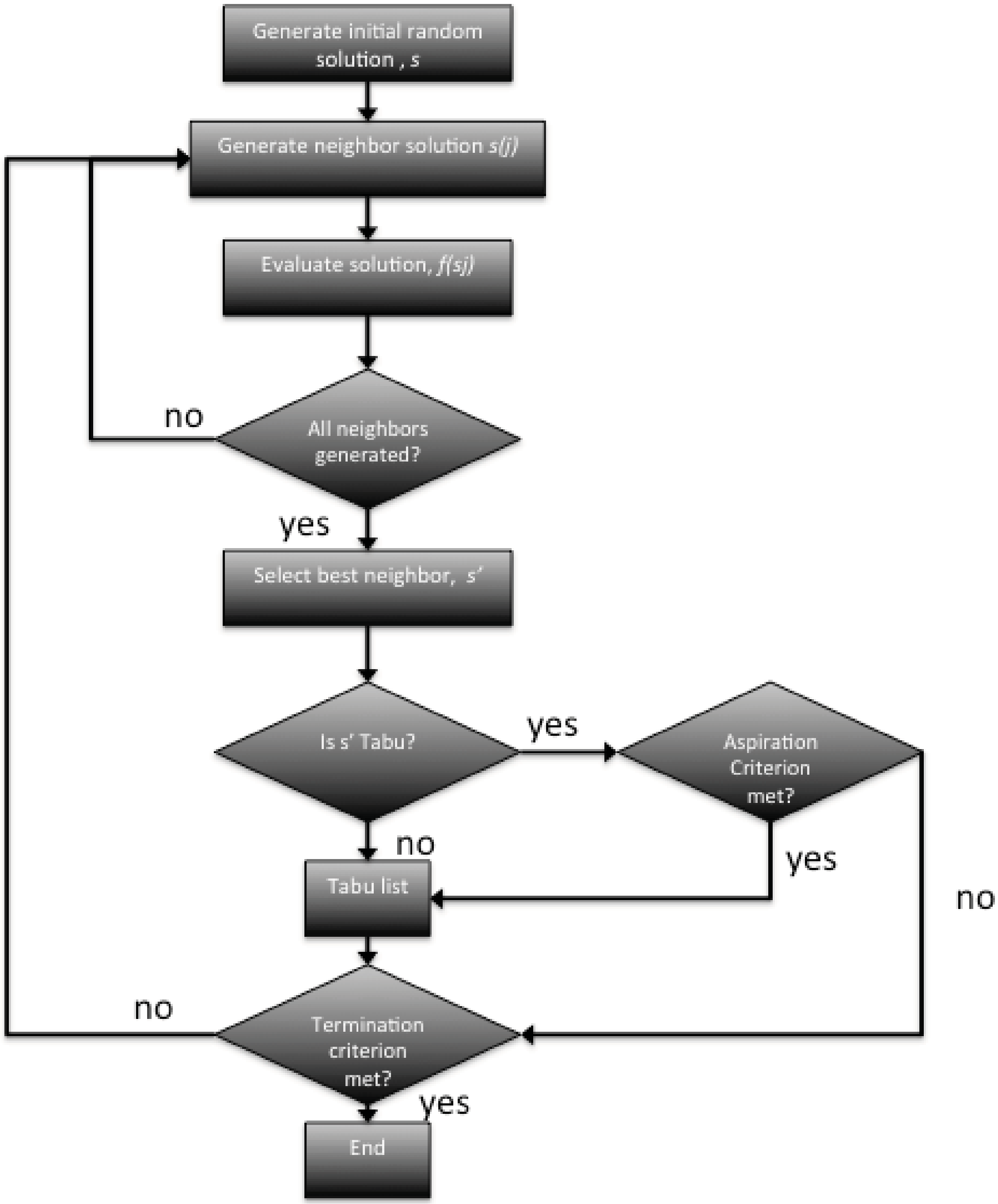


Figure 2-6: Tabu Search Flow Chart

2.4.3 Ant Colony Optimization

Marco Dorigo proposed ant colony optimization for the first time in 1992. . In his thesis Dorigo uses Ant Colony Optimization to search in a graph the optimal path. Ant Colony Optimization is the metaheuristic inspired in the behavior of real ants, and is part of the swarm intelligence, and it is based on the foraging behavior of ants seeking a path between the colony and the source of food. At the beginning ants search the area surrounding them at random, and when they find a source of food they creates a path to it. The path is created when at the return trip the ants deposits a chemical pheromone on the ground. As better the quality and quantity of the food more pheromone is deposited on the ground, and then more ants are guided to the source of food. By using it pheromones the ants can find the shortest path to the best source of food. Ant Colony Optimization uses this characteristic to solve hard optimization problems. Ant Colony Optimization has a good performance in scheduling problems, as Job-Shop, Flow-Shop, Resource Constrained Project (Besten *et al*., 2000), Single Machine Total Weighted Tardiness problems (Dorigo *et al.,* 1996), and Traveling Salesman Problem (Merkle *et al*., 2002).

Ant Colony Optimization is a metaheuristic that has an initialization step and then it loops over three algorithm components. One iteration loop consists in the construction of all the solutions by the ants, and then in the second loop they are improved by the use of a local search algorithm, then finally in the third loop an update of the pheromones is done. To construct an set solutions, *m* artificial ants construct the solutions by using the elements form a finite set of solution components, $\mathbf{C}=\{c_{ij}\}$, i=1…..n, j=1,….,D_i. In a empty partial solution $s^p=0$, the construction of initial solutions starts, and then the current partial solution s^p is extended by adding a component of feasible neighbors. The component chosen from the neighbor is done in a probabilistic manner at each construction step. The rule for the choice of solution components is the following.

$$\mathrm{p}(c_{\mathrm{ij}}|s^p)=\frac{\tau_{ij}^{\alpha}\ .\ \eta(c_{ij})^{\beta}}{\sum_{c_{ij}\,\epsilon\,N(s^p)}.\ \tau_{ij}^{\alpha}\ .\eta(c_{ij})^{\beta}}\ , \qquad \forall\ c_{\mathrm{ij}} \in\ N(s^p),$$

Where τ_{ij} is the pheromone value associated with the component c_{ij}, and $\eta(.)$ is a function that assigns at each construction step a heuristic value to each feasible solution component $c_{\mathrm{ij}} \in\ N(s^p)$. Alpha and beta are the parameters that determine the importance of the pheromone, and are always positive. Once the first loop is finished the next step is to use a local search to optimize locally the solutions and then finally update the pheromones. The pheromones increase the values associated of the "good" solutions and decrease the values of the "bad" solutions. There are to ways in order to update a pheromone one is to decrease the pheromone values by pheromone evaporation and the second is to increase the pheromone levels of the "good" solutions. The most commonly way to update a pheromone is by increasing the value of the "good" solutions (S_{upd}) and is done using the following formula:

$$\tau_{ij} \leftarrow (1-\rho).\,\tau_{ij} + \rho. \sum_{s\epsilon S_{upd}|c_{ij}\epsilon s} F(s)\ ,$$

where S_{upd} is the set of solutions that are used for the update, $\rho\ \epsilon$ (0; 1) is a parameter called evaporation rate, and F(s) is a function such that $f(s) < f(s') \Rightarrow F(s) \geq F(s')$, $\forall s \neq s'\ \epsilon\ S$. The pseudo code of the procedure is in figure 2-7.

```
Initialize the base attractiveness, τ, and visibility, η, for each edge;
for i < IterationMax do:
        for each ant do:
        choose probabilistically (based on previous equation) the next state to move
        into;
        add that move to the tabu list for each ant;
        repeat until each ant completed a solution;
end;
for each ant that completed a solution do:
update attractiveness τ for each edge that the ant traversed;
end;
if (local best solution better than global solution)
save local best solution as global solution;
end;
end;
```

Figure 2-7: Ant Colony Pseudo Code

2.4.4 Monkey Algorithm

Monkey algorithm is one of the most novel evolutionary algorithms presented. It is inspired in the mountain climbing process of the monkeys. For the novelty of its implementation in combinatorial optimization problems there is no past research that gives a brief idea in which circumstances the Monkey Algorithm works best. Monkey Algorithm consists id three major steps (called processes): the climb process, the watch jump process, and the somersault process. The climb process tries to simulate the monkey's mountain climbing process. The climb process is a process to search a local optima using the pseudo-gradient information of the objective function.

The watch-jump process is when a given monkey is at the top of the "mountain". When this happens the monkey will look around in order to find higher mountains to climb. When a higher mountain is found the monkey jump to it and the climb process begins again. To do this neighbor solutions are generated at random in order to find better solutions. Finally the somersault process is the process of taking the center of all the monkeys as a pivot, and then each monkey will somersault to a

new position. This position could be forward or backwards the direction of pointing at the pivot, and then at new positions the monkey can continue climbing. After the stopping criterion is met the monkey that climbs the highest mountain is the monkey that finds the optimal solution. One advantage of a Monkey Algorithm is that it can solve different optimization problems including problems where the objective function is neither differentiable nor linear. Another advantage is that Monkey Algorithm has a limited number of parameter, which makes it very easy to implement.

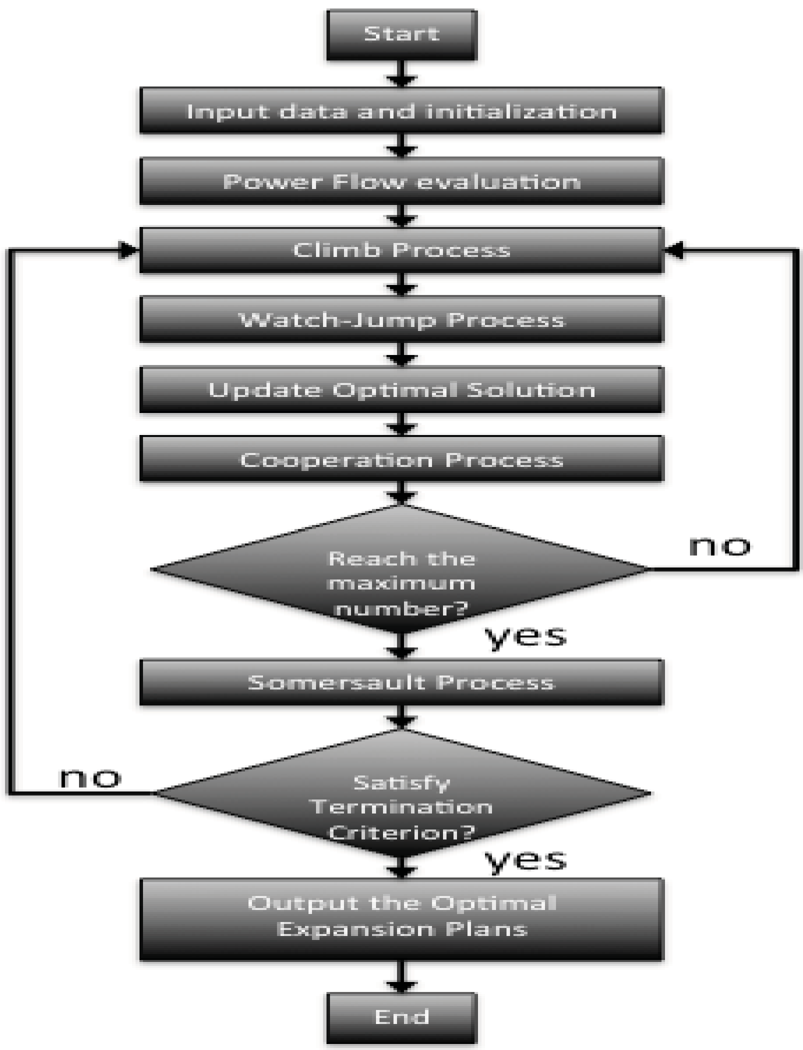


Figure 2-8: Monkey Algorithm Flowchart

2.4.5 Genetic Algorithm and Memetic Algorithm

John Holland introduced the Genetic Algorithms in 1975, based on the theory of evolution. Genetic Algorithms along with the Memetic Algorithms belongs to the larger class of Evolutionary Algorithms. Genetic Algorithms and Memetic algorithms are search algorithms based on natural selection and natural genetics, and generate the optimal solutions using techniques inspired in the natural process of selection of individuals and the evolution of species. The techniques used by the Genetic Algorithm tries to imitate the reproduction mechanism and the genetic transmission of characteristics. This is an example of how the new species are originate by natural mechanisms to outside the other species that are not well adjusted to the environment.

Genetic and Memetic Algorithms are in the category of global optimization methods. Both algorithms are global optimization methods because the trial solutions run are generated based on the global information obtained during the search process. Generally speaking, the combination of evolutionary algorithms (EAs) with local search gave birth to "Memetic Algorithms" (MAs) (Moscato, 1989). Often, classical GAs are not aggressive enough to solve combinatorial optimization problems and should be enhanced with local search methods (Mendoza *et al*., 2009). The role of local search in Memetic algorithms is to locate the local optimum more efficiently than the Genetic algorithms (Garg, 2009). It was in the late 80's that the term 'Memetic Algorithms' was adopted to denote a family of metaheuristics that blended several concepts from tightly separated –at that time– families such as evolutionary algorithms (Moscato *et al*., 2004). According to Goldberg (1989) the genetic algorithms are different from other metaheuristics in four different ways:

1 Genetic Algorithms do not use deterministic rules instead GAs uses probabilistic rules.

2 Genetic Algorithms uses the objective function information.

3 Genetic Algorithms works with a coding parameter set.

4 Genetic Algorithms search space is a set of population points.

Since one of the main difference between Genetic/Memetic algorithms is the representation one of the most important steps before implement them is to fully understand its encoding for solutions. The solutions in a Genetic/Memetic algorithm are presented in **"chromosomes"** (sometimes also called genome), which is a representation of a string that has a determined length where the complete information of the given solution is stored. It is important to denote that even though the nature uses multiple chromosomes, Genetic/Memetic Algorithms only uses one chromosome to encode all the information of a solution. Each chromosome is form by the union of many genes, where a gene is the smallest information unit, and it is represented using one type of data.

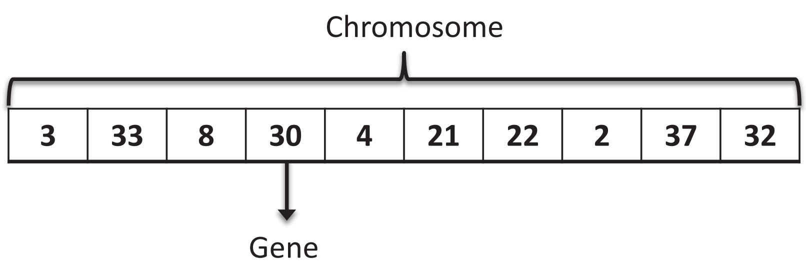


Figure 2-9: Representation of a chromosome and a gene

Figure 2.9 shows the representation of a chromosome with a permutation encoding, where the gene can be represented in integer numbers. In order to solve a problem using Genetic/Memetic algorithms is indispensable to define five elements:

1 Select the representation of the chromosome for the problem (select encoding)

2 Select a method to find a initial population (a set of chromosomes).

3 Select the formulas for evaluate the given solution/chromosomes (fitness function).

4 Select the parameters for the genetic operators (which will modify the genes in the chromosomes). *For the Memetic Algorithms select the local search to be used.

5 Select the values of the parameters (for example population size).

In order to use Genetic/Memetic algorithms the first step is to select the characteristics of the chromosome used. The main parameters for a chromosome will be the length of the string and the type of encoding. While the length of the string can be flexible for a given problem, the type of encoding must me carefully selected. The length of the string is dependent to the size of the given problem selected, but in the other hand the type of encoding is depended of the nature of the problem. There are three main types of encoding used for Genetic/Memetic Algorithms: the binary encoding, the integer/permutation encoding and the real-value encoding.

Binary encoding uses the values of bits (0 or 1) to represent the information in a gene. The use of binary encoding has the advantage of giving a big set of possible chromosomes with a very natural and direct representation. For combinatorial optimization problems and other types of optimization problems being solve using Genetic/Memetic the binary encoding is the most common type of representation.

1	**1**	**0**	**0**	**1**	**1**	**1**	**0**	**1**	**0**

Figure 2-10: Example of a chromosome with binary encoding

Permutation encoding uses integer numbers to represent the information of a gene; most of the time the that number represents a value in a sequence. Permutation encoding helps to search for a best combination of items that are subject to constraints. Permutation encoding is best used to solve combinatorial optimization problems and ordering problems such as the Traveling Salesman Problem or a Task Ordering Problem.

1	**5**	**3**	**1**	**6**	**8**	**2**	**9**	**4**	**7**

Figure 2-11:Example of a chromosome with permutation encoding

Real-value encoding uses a specific value at each gene; this values can be anything, from numbers to letters or words. Real-value encoding is used in problems where complicated values are used, but especially when the use of binary of permutation encoding is very difficult. Real-valued encoding is most commonly used for function optimization problems. Some examples of real value encoding problems are: Trees, schedules, tours, and other combinatorial problems can easily be represented by using real-valued vectors.

Chromosome A	**5.3**	**3.1**	**6.1**	**2.6**	**3.83**	**0.32**	**10.9**	**8.547**	**1.007**
Chromosome B	**A**	**H**	**T**	**U**	**K**	**D**	**A**	**X**	**G**
Chromosome C	**Right**	**Center**	**Left**	**Back**	**Front**	**Center**	**Back**	**Right**	**Left**

Figure 2-12:Example of a chromosome with real-value encoding

To create a Genetic/Memetic algorithm in practical problems the following six steps must me followed:

1 Initialization of random population

2 Fitness function

3 Selection

4 Genetic Operators
4.1 Local Search (for memetic algorithms only)

5 New population

6 Termination Criterion

The first step to use a Genetic/Algorithm is to generate a set of individual solutions that must be generated at random, where is individual of the population is a solution for the problem, even though

they aren't the best solutions. The size of the initial population will be determined by the nature and size of the main optimization problem, it could form hundreds up to thousands of solutions. The fitness function is the formula that will evaluate the quality of the chromosome, by using this quality value the Genetic/Memetic algorithm will chose if the chromosome will continue the process of reproduction.

The selection operators have the same role than the natural selection where the overall effect is to bias a gene to following generations. Those genes are the genes, which belong to the most fitted individuals at the current generation. The chromosomes chosen to evolve where the chromosomes in the initial population that after evaluation of fitness value got the highest score. Sometimes it is important to do not chose at each iteration all the chromosomes with the highest value in order to avoid a premature convergence. There are different selections schemes to be used in Genetic/Memetic Algorithms such as Elitism, "Roulette Wheel" selection, Rank Selection, and finally Tournament Selection.

The elitism selection selects the best fitness value chromosomes of the entire population. By doing this the selection tries to select the better individuals or at least the individuals the are biased to be the best ones. This selection is done by choosing the best-fit value chromosome or a small set of chromosomes, and using them in the new population. The other values needed to complete the size of an initial population are then selected again by random. Elitism is one of the more used selection criterions when using Genetic/Memetic algorithms because it allows to the solutions to go better over time and by doing so increase the performance of the algorithm, also another important factor of the elitism is that it prevents to loose the best found solution.

In roulette wheel selection the best-fit chromosomes are selected to be the parents chromosomes used by the genetic operators. The chances of selection are directly correlated to the fitness value, such that the better the solution the higher probability of being chose as a parent chromosome. Roulette wheel selects potentially good solutions for the recombination or reproduction. Using roulette wheel all

chromosomes of the population has the probability of being selected accordingly to its fitness function. Roulette-wheel has the limitation that when one chromosome has very good fitness value the other chromosomes may have any real chance to be selected as parent chromosomes in the next step of the Genetic/Memetic algorithm process.

Rank selection process is similar as the roulette-wheel process, but it eliminates the main disadvantage of the roulette-wheel selection. Rank selection first ranks the population and then the chromosomes are again ranked according to it fitness value. For example, the chromosome that has the lower fitness value chromosome is going to have the fitness value rank number one, then the second lower fitness value chromosome will have the fitness rank number two, until the best fitness value chromosome have the least fitness rank. By doing this all the chromosomes will have a real opportunity to be selected as parent chromosomes. One of the main problem of this style of selection is that it leads to a slower convergence, because the best chromosomes will not differ a lot from one another. The last type of selection is the tournament selection. Tournament selection run "tournament" for few individuals (chosen at random), where the winner of each tournament is the parent chromosome selected.

The fourth step to use a Genetic/Memetic Algorithm is the use of the genetic operators, and in the case of Memetic Algorithms also the use of a local search. Genetic operators use the selected parent chromosomes to be modified its components in order to explore the decision space. There are two main genetic operators the crossover operator and the mutation operator. The crossover operator exchanges the information of two parent chromosomes to produce a new chromosome that has the information of both parents.

The main objective of using a crossover operator is to obtain a better chromosome by using the information of two chromosomes (specially when the two parent chromosomes are good value fitted chromosomes). There are different types of crossover operators; one is the "one point crossover", where

a random gene is selected to be crossover point. In one point crossover two “children” chromosomes will result from the exchange of information of the parents, one that will have all the information before the gene selected of the first parent and the information after the gene selected of the second parent and the second child will have the rest of the chromosomes in formation.

Another example is the two-point crossover operator where again at random two gene points are selected and all the information that is in between those two points will be exchanged by the parent chromosomes. Those were the most widely used crossover operators, but crossover could be done when two chromosomes exchange information to create a new chromosome that has some information of both. Another type of crossover operator is when the none-genes of a parent chromosome join the even-genes of another parent chromosome to form a new “child” chromosome as is shown in Figure 2.13

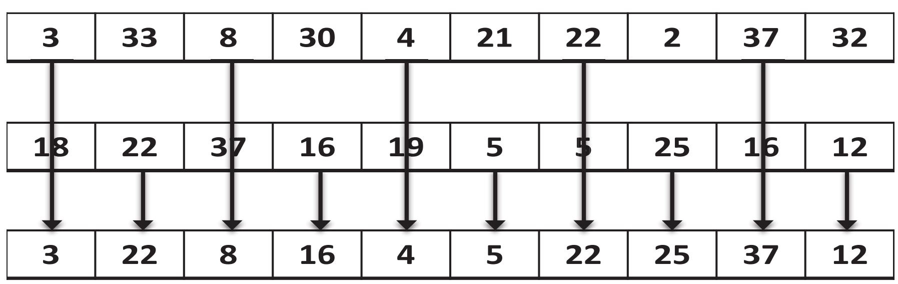


Figure 2-13: Example of a crossover operator

The genetic operator changes at random one or more genes in a chromosome, giving birth to a new chromosome to be evaluated. A change in a gene value in a good fitted chromosome helps the genetic algorithm to arrive at better solutions. As the crossover operator the mutation operator helps the genetic algorithm to stall at a local optima, the mutation operator is defined by a mutation probability established by the user. Most of the time the mutation probability falls between 1% and 10%.

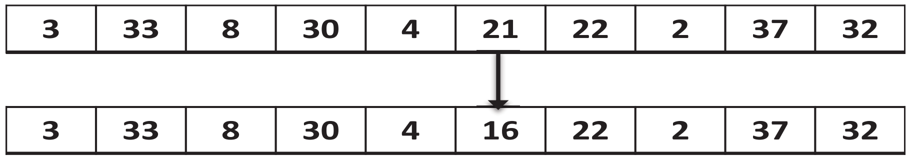


Figure 2-14: Example of a genetic operator

For the case of a Memetic algorithm another operator, a local search, must be added to the search process. Local search algorithms use one chromosome and then move it to a neighbor search space solution chromosome. Most of the time chromosomes has a vast number of neighbor solution chromosome and indoor to chose to which neighbor move is taken by the information of the neighbors. There are different local search metaheuristics such as simulate annealing or Tabu search. Memetic algorithms combine a Genetic Algorithm with any type of local search, to increase the search space at each iteration.

The fifth step needed to use a Genetic/Memetic algorithm is the creation of a new population. In this step the best-fitted chromosomes found after the genetic operator, and in the case of the memetic algorithm, also the local search are selected to be the population used in the next iteration. Most of the time 30% to 50% of the new population is based on the best chromosomes found in the last iteration, the rest of the population is set at random. This is step is important to ensure that the best solution are maintained until the end of the Genetic/Memetic algorithm process but at the same time it ensures the diversity of solution by setting a percentage of the population at random.

The last step to run a Genetic/Memetic Algorithm is to set the termination criterion. The termination criterion states when the Genetic/Memetic algorithm stops to run, it is the maximum number of generation. Sometimes a termination criteria is when after a certain specific consecutive generations the same optimal solution is found. In short to run a Genetic/Memetic algorithm the parameters to be

specified are population size, mutation and crossover probability, the selection method and the termination criterion.

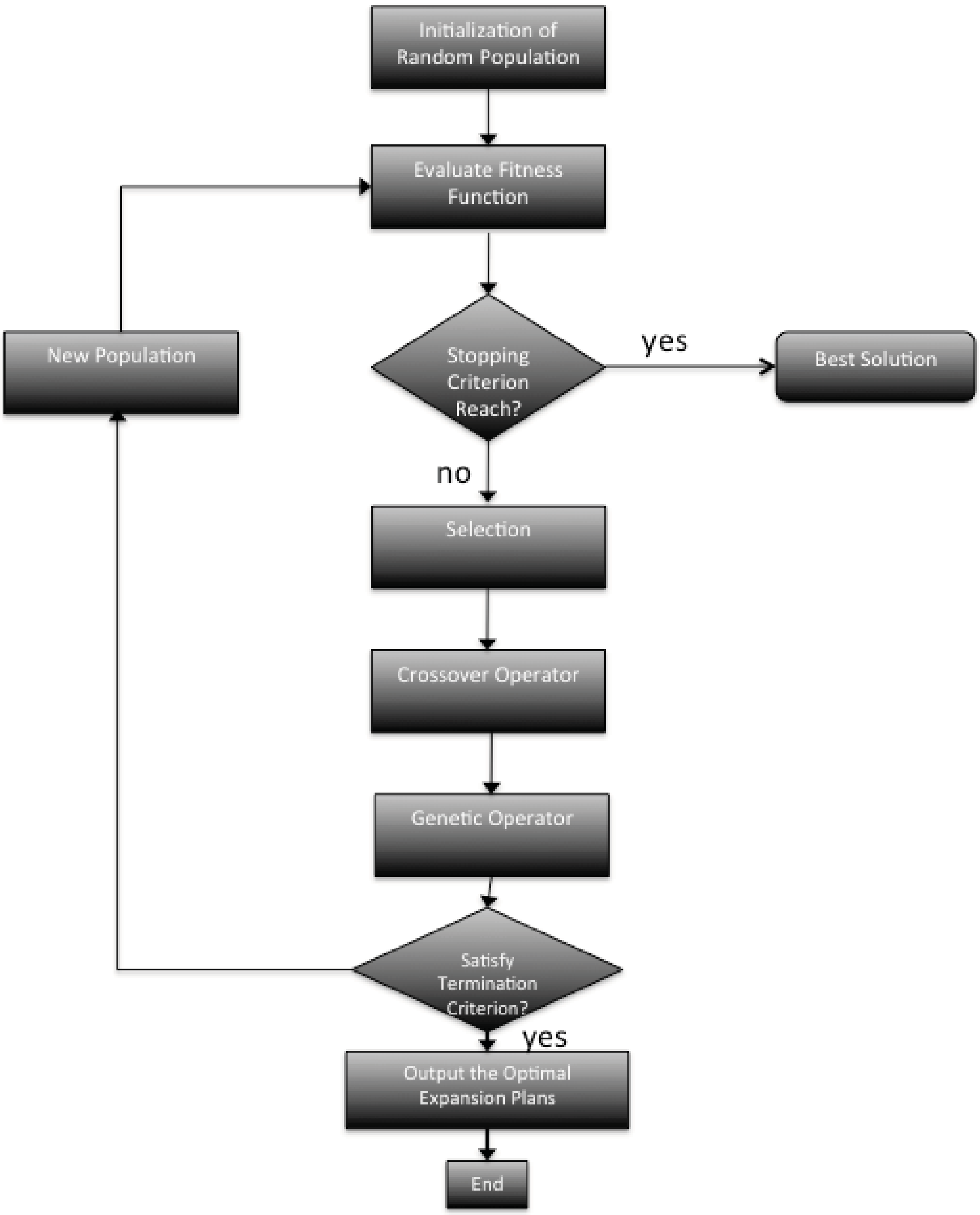


Figure 2-15: Genetic Algorithm Flowchart

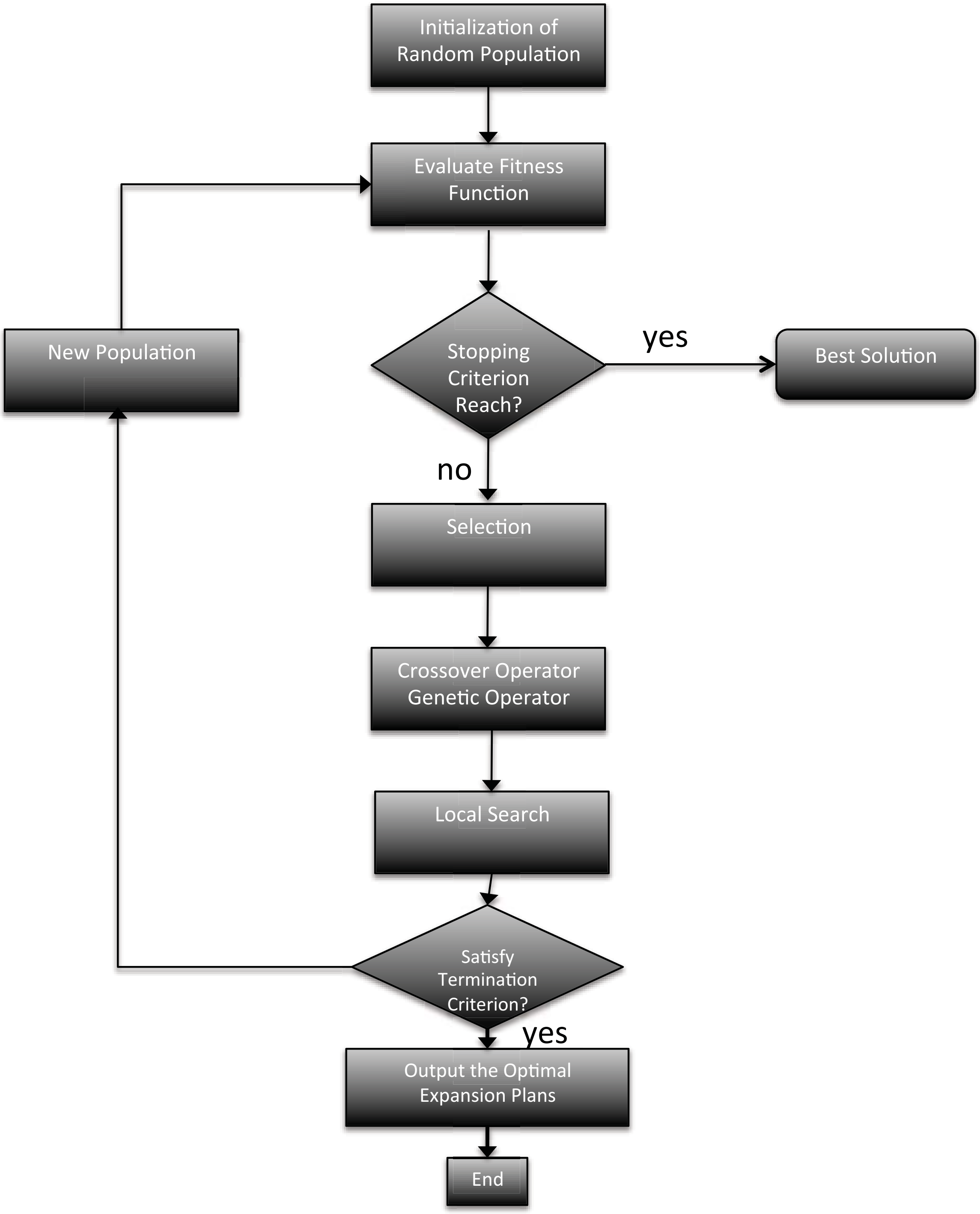


Figure 2-16: Memetic Algorithm Flowchart

2.5 Conclusion

Chapter 2 presented a definition for a combinatorial optimization problems and two solution methods were proposed. Exact optimization and metaheuristics were explained in detail. Evolutionary methods such as Ant Colony Optimization, Particle Swarm Optimization, Monkey Algorithm, Genetic and Memetic Algorithms were explained. Especial emphasis was put on explaining to explain the working, formulation and methodology of Genetic and Memetic algorithms, since both are the tools used in the present research to solve the power distributions system replacement scheduling problem and the aviation baggage screening security problem.

Chapter 3: Component Replacement Analysis and Power Systems

3.1 Component Replacement Analysis

For economic decision models such as maintenance and replacement decisions, replacement analysis is the most useful tool. Component replacement tries to solve the problem of "if and when" a component should be replaced in configuration where the system is always in efficient working conditions. Since we are in a time where the technology is constantly changing and some of the equipment used is becoming old, determining the optimal way to replace the components used in different industries is becoming one of the main research topics. All of assets that provides a service or produce a product wear out over time and for that its output lower as time passes. Everything could be replaced form tool, machines up to buildings, roads and bridges, and since the replacement of them is inevitable the main question to solve is when.

Bellman (1955) formulated the component replacement problem as a dynamic program, proposing the optimal replacement policies first. The main disadvantage of this component replacement schedule is that it does not takes into consideration the technological change neither any technological improvement. Sethi and Chad (1979) also used dynamic programming to solve a component replacement problem, this time they considered the change and improvement of technology over a finite time horizon. The component replacement designed by Sethi and Chad minimizes costs, maximizes profit and minimizes the number of breakdowns over a finite time horizon. In 1991, Brown developed a serial replacement model that considers all the risks, trying to minimize the impact of the decision taker on the minimization objective when the utilization of an asset and its utility function is unknown. Brown developed a dynamic programming model to help the decision maker to choose over a set of "optimal solutions" his preferred alternative for the replacement schedule. The main problem of the last component replacement techniques s that they focus in a single machine problems, but in real-life all the component replacement schedules must include different types of machines that conforms a hole system,

such an industry. Leung and Tanchoco in 1990 developed a model that divided the production systems in a set of networks, and then do a replacement analysis for each network. This model structure is a multi-stage problem that has a set of multi-commodity flow of sub-problems per stage, and it is formulated as linear programs solved using dynamic programming. Also, Esogbue and Hearnes in 1998 used fuzzy sets to model the uncertainty of the equipment to calculate the economic life different components over a finite time horizon, and then make more accurate decisions.

Most of the time the component replacement problems are solved over large time horizons, with the replacement decision taken at specific points of time, but the production planning solutions but need to be addressed per week or per day. In 2000 Hartman used integer programming to solve a component replacement problem where the number of components required for a task that considers a fixed and a variable replacement cost. Hartman also introduced in 2001 a stochastic dynamic programming formulation to make a replacement scheduling assuming a non-continuous utilization of the components. Hartman states that the wear-out of a component is not only influenced by its age but also for its utilization, changing the problem from deterministic to probabilistic. Finally in 2004, Hartman developed a dynamic programming model that finds the optimal replacement schedule for different components over a finite plan horizon assuming a stochastic demand.

In 2005 Childress and Cohen presented a dynamic programming program that solves an stochastic parallel machine replacement problem. Childress and Cohen divide the parallel machines as a set of independent Markovian processes. As it can be seen most of the component replacement problems are solved using dynamic programming, but dynamic programming has the disadvantage that when the problem becomes large the state space becomes enormous, so that the efficiency of the metaheuristic is compromised. Yano (1984) suggested a series of techniques that contributes to the development of better solutions for component replacement. In 1994, Karabakal *et al*; presented a replacement problem where the cash flow of a future asset is not dependent on it service conditions neither the previous

replacement decisions. Here the replacement schedules are determined such that the net present value of the cash flow resulting from the schedules is optimized.

In 1991 Hoppr and Nair solved the component replacement problem where they forecast the technology changes, and those changes and the revenue and cost functions are not stationary on time. Nair and Hoppr presented a continuation of this work in 1992, where they solve the component replacement problem using dynamic programming. Rajagopalan in 1998 developed a deterministic model that allows the replacement of capacity and adaptability to demand changes. Later in 1999 Létourneau *et al;* developed a component replacement schedule for aircraft industry using data mining. They develop an schedule that uses the data obtained by censor to predict the failure of a component and then replace it just before it fails. In short it can be said that an asset should be replaced when it cannot be repaired, when is no economically feasible to continue using it or when new technology has surpassed the output of asset at optimal conditions. Sometimes the decision to change is taken only because there is a desire of continues improvement. The main reason that makes component replacement important is because it helps to provide a required service over a predetermined planning horizon in a economical efficient way. Replacement analysis takes the decision of whether or not replaces an existing asset with a new one. The output/performance of a given asset decreases a time passes, also the maintenance cost increases as time passes. For those reasons a timely replacement for those assets is important, in order to have an efficient operation.

To determine the best component replacement schedule an analysis of the present and future cost of the components. Most of the component replacement analysis focus only in the decision of either to KEEP or REPLACE the component, assuming that a constant maintenance is done to the components. Component replacement analysis determines the best sequence of KEPP/REPLACE components in order to maximize the output, but most important minimize the total cost. Alchian in 1953 was the first to define the costs related to provide a service, such as the current value of the equipment, the cost

associated of buying new equipment, operating costs, all of them based on a present value of the costs. The most common costs involved in component replacement analysis are: purchase costs, maintenance cost and the cost of unmet demand.

3.2 Electric Power Systems Power Systems

The first objective of the Electric Power System is to satisfy the customers electric demand as economically and reliable as possible with a good quality and continuity (Billinton and Allan, 1998). Electricity is obtained by the conversion of some sources of energy such as nuclear, solar energy and coil, which are called “primary sources”. There are two types of primary sources: renewable and not renewable. Electricity is the type of energy most commonly used and it is continuously expanding this dominance. In 1940, only 10% of the energy used in the United States was electricity but by today it represent the 40% of energy consumption in the country (U.S. Department of Energy). The power grid is divided in three: generation, transmission network and distribution network (in order).

Electric energy is first produced at low voltages, and then it is transformed to higher voltages (for a more efficient long-distance transmission). High-voltage power lines then compose the transmission network, and then the electricity is transported in bulk over long distances. When the electricity arrives to the point where it is going to be needed the voltage is lowered again at a local sob-station (the distribution network). The distribution system is final link between the bulk transmission and the final user. The distribution system is most commonly considered from the substation to the customer meter. Almost 80% of the customer electrical interruptions occur for failures in the distribution system (Chowdhury and Koval, 1998). The first half of this research is dedicated to solve a component replacement scheduling for the power distribution system. One of the major problems of the power systems is that it is vulnerable to control failures, communication failures and disturbances (Liu *et al;* 2000).

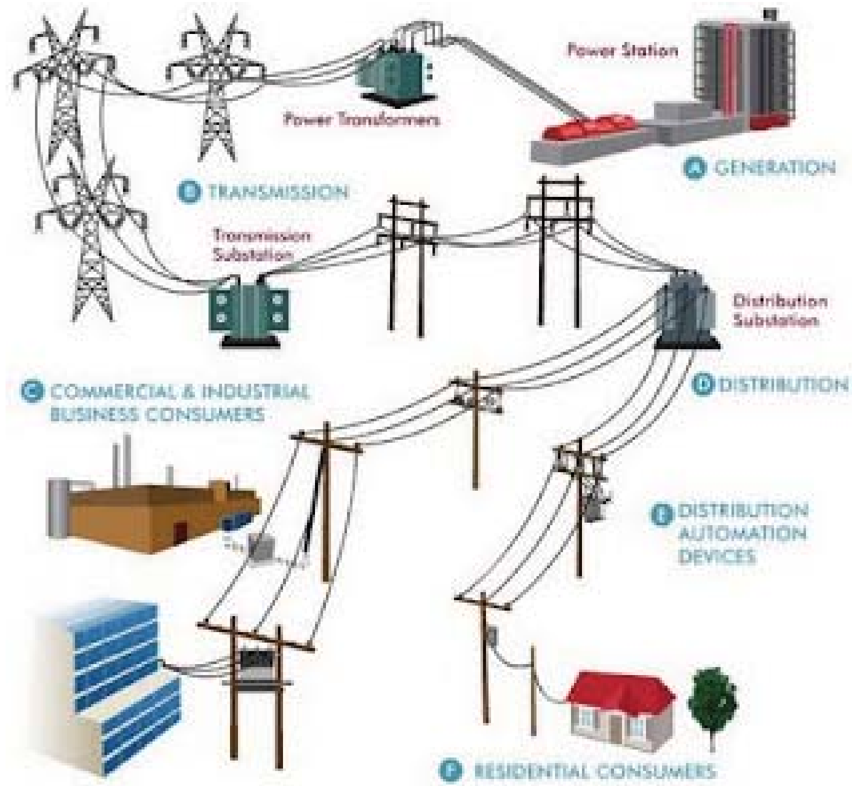


Figure 3-17: Electric Power Grid

3.2.1 Electricity Generation

The electric industry begins in the early 1880's when companies begin to use the water-driven generation of electricity for the operation of street lightning, this was the first application of electricity in the United States. Power plants transform sources of energy in electricity by the use of electro-mechanical generators. Most of the electro-mechanical generators work with heat engines that use as fuel the flow of water, wind, chemical combustion or nuclear fission. Those are the most common sources of energy, but they are not the only ones, also exist the use photovoltaic cells and the geothermal energy. The generating plants first used the fuel energy to convert it into heat and then create steam to make work the electrical generators, and then the electricity is transmitted by high tension transmission

lines (Pansini and Smalling, 2002). Almost all of the electricity generated is produced by the use of fossil fuels. The fossil fuel is burned to create heat and then produce steam to make the generators work.

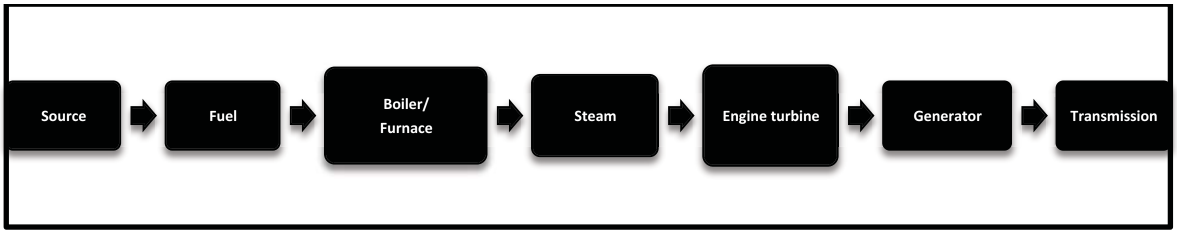


Figure 3-18: Energy Conversion

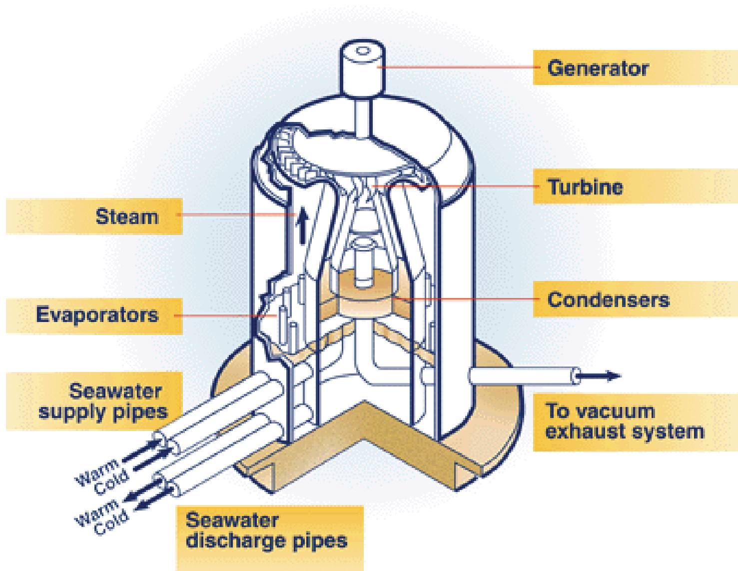


Figure 3-19: Electricity Generator

The most common fossil fuel used to generate electricity is coal, while natural gas is also commonly used to generate electricity. As coal, natural gas is burned to heat water and produce steam to make the generators work, but for the high prices of gas it is most commonly use when there exist a high demand. As natural gas and coil, petroleum can be burn to heat water and create steam to make the turbines work. The main disadvantage of using coil, natural gas and coil is that them cannot be replaced fast because it takes thousands of years to form them naturally. Another important source for the

conversion of energy to electricity is the Nuclear energy. Nuclear energy heats the water to produce steam by nuclear fission.

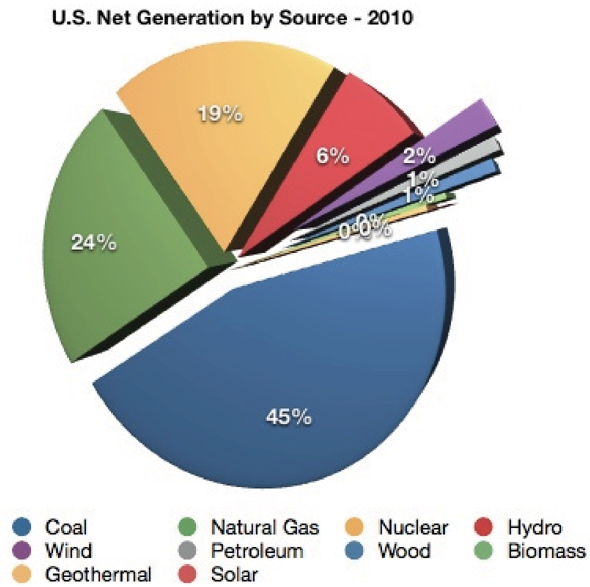


Figure 3-20: U.S. Electricity Generation by source 2010 (source mapawatt)

There exist two main types of electricity generation plants: the most commonly used plants that generate electricity from non-renewable, nut more economic, sources and the plants that generate electricity from renewable sources. In the first case the generation of electricity can be divided in three major areas:

1. Steam Turbines
2. Combustion Turbine
3. Reciprocating Engines

For the steam turbines to work water must be heated to produce steam and then the steam moves the turbines to generate electricity. One way to generate steam is to burn water to produce steam is to use fossil fuel such as coal, oil and natural gas. The other way to heat water is by using nuclear energy. Nuclear plants uses a reactor to heat water, where a nuclear fuel (most of the time uranium) is heat by

neutrons and creates fission that releases heat and more neutrons. Those neutrons heat more uranium atoms and so on. This heat generated is used to convert water into steam and make the turbines generate electricity.

The combustion turbines most of the time uses as fuel gas. In combustion turbines the gases are burn and then expanded. This expansion makes the generators blades move and then generate electricity. One key component of the combustion turbines is the compressor that increases the pressure by a factor of ten. By using the compression of air the temperature obtained is increased, making the expansion of the turbine more effective. The reciprocating engines are engines that use fuel as a source of energy (most of the time diesel). As the other methods to generate electricity the output shaft is connected to an electrical generator.

The other form to generate electricity is by using the renewable energy such as hydropower energy, wind energy, biomass energy, geothermal energy and solar energy. Hydropower energy is the most widely used form of renewable energy, since it uses the gravitational force over the water to move the electricity generators. Hydropower does not has direct waste and produces almost zero greenhouse emission compared to all the non-renewable sources of energy (Kabisama, 1993). Wind energy is made by the conversion of wind into electricity. This is achieved using wind turbines, which uses mechanical energy to generate electricity. Biomass is a fuel obtained from plants and animals such as food scraps, grass, and municipal solid waste. These fuel is burned to produce steam to move the blades of the electricity generators. The geothermal is the heat energy stored in beneath the surface of the earth, as most of the ways to generate electricity, geothermal uses the heat to convert water into steam and move the blades of the electricity generators. Solar energy is the conversion of sunlight into electricity using photovoltaic and solar-thermoelectric plants. PV cells converts sunlight directly into electricity while solar-thermoelectric plans uses the sunlight to heat water and produce steam. After the electricity is generated the next step is to move it to the final user, this is done in the transmission network.

3.2.2 Transmission Network

The transmission network interconnects the electric generation power plants with the distribution system. One of the main characteristics of the transmission systems is that they are designed to transport the electricity at the speed of light because there is no capacity for long-term electricity storage. The transmission network carries alternating current, direct current or a combination of both, and all the transmission lines can carry three-phase current. Most of the time electricity is transported in high voltages in order to minimize the lost of energy. Transmission lines can be overhead, underground or submarine.

All the AC transmission lines share the characteristic of being three-phase current, where the voltages changes depending on the grid where they belong. All the voltages vary form 69kV to more than 800kV. The overheard transmission lines are classified according to the voltage that they carry, and there are five different classifications: low voltage, medium voltage, high voltage, extra high voltage, and ultra high voltage. Low voltage carries less than 1000 volts and is used for connection to residential use. The medium voltage is between 1000 volts and 33000 volts and it used for distribution of electricity. High voltage carries a voltage form 33000V to 230kV and it is mostly used for transmission of bulk quantities. The extra high voltage and the ultra high voltage transmission lines are used for long distance energy transmission, for the extra high voltage transmission the voltage is form 230kV up to 800kV, and the Ultra High Voltage transmission are for voltages higher than 800kV.

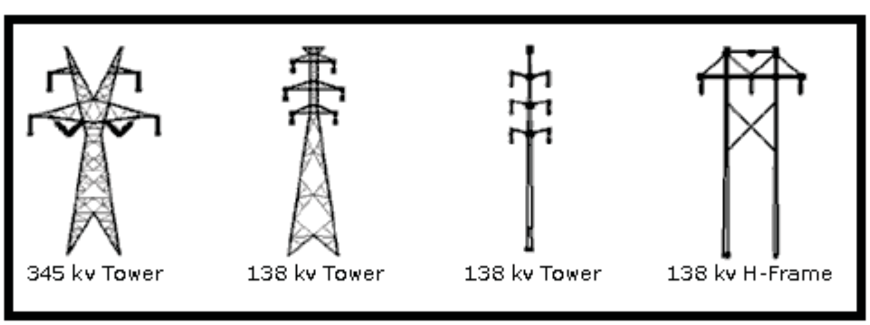


Figure 3-21: Example of overhead transmission lines

The underground transmission lines are used in high density such as big cities, and they are an alternative for the overhead transmission lines. Underground transmission lines are also used in areas where there are natural obstacles to use the common overhead transmission lines. One of the main disadvantages of the underground transmission lines is that as the voltage increases a grater protection must be also placed, for example construct tunnels for the transmission lines. The main reason for this protection is because as electricity passes thru the transmission cables begin to generate heat, so that a cooling system sometimes is installed. Another disadvantage of the use of underground transmission lines is that the maintaining and repair time is higher compared to the maintaining and repair time of the overhead transmission lines. In the other hand the underground transmission lines have some advantages over the overhead transmission lines, for example the weather has almost zero impact over them for they nature if being under the constructions of the cities, lowering the possibility of a blackout for any natural disaster.

Another advantage of using underground transmission lines is that they reduce the emission electromagnetic fields because the earth surrounding them reduces the power of the fields. Underground transmission cable has only two different categories: the cables with les than 70kV (distribution cables) and the cables than can carry a voltage range from 69kV up to 500kV (more commonly known as the high voltage power underground transmission cables). The last type of transmission lines is the submarine transmission line. Submarine transmission lines are commonly used for high voltages because water is a natural cooling, and the pressure created by water also reduces the possibility of void formation. The most common used submarine transmission cable has cross-linked polyethylene insulation and an aluminum alloy that is resistant to corrosion. The manufacture of the submarine power transmission lines provides them a high tensile strength and allows the installation in deep water. One disadvantage of the use of submarine transmission cables is the environmental impact for fluid leaks.

Figure 3-22: Underground Transmission line

Figure 3-23:Underground Transmission Cable

3.2.3 Distribution System

Once the electricity arrives the cities it must lowered on voltage at a substation, and then the electricity can go to the homes and factories through a distribution system. The main function of the distribution network is the connection between the bulk power systems with the user al low voltages. The distribution system has the greatest impact on the consumer reliability (Billinton and Wang, 1998). The distribution system is the power system commonly seen by the customer, and for that it plays the largest role for quality service perceived by the customer. A part of the distribution systems are the poles and wires seen across the streets of a city. The distribution system has four primarily components: a

distribution substation, a primary feeder, a distribution transformer, and secondary lines and services. In normal distribution networks there are several components such as transformers and primary distribution feeders.

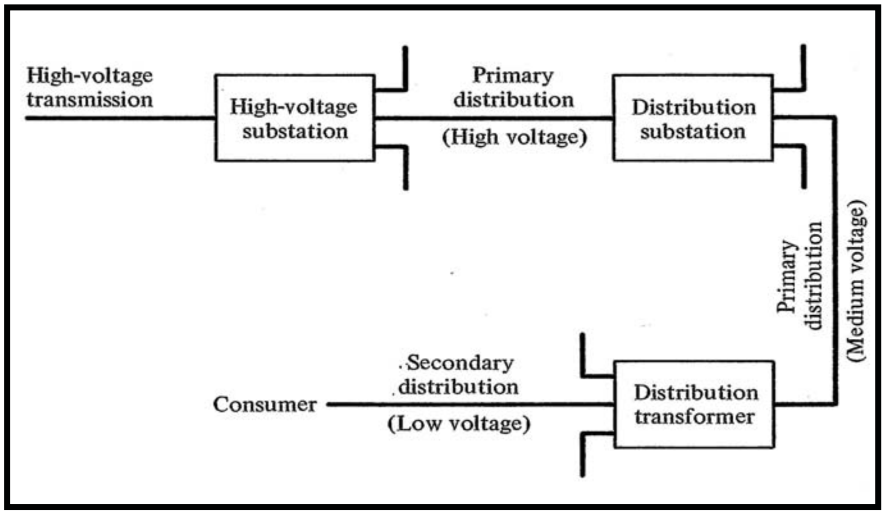


Figure 3-24: Electric Distribution System

Source: Electric Power Transmission Systems, Eaton & Cohen, (1983)

The most common voltage used in utilities in the unites states is 13kV, but it has been replaced in recent years for 4kV, while in the most dense areas the primary voltage used is 34kV. The voltage is lowered at a substation. High-voltage electric facilities are called substation. A substation switch generators circuits, lines and equipment out of a system, changes alternate current to direct current, and most important changes the voltage of any alternate current charge. The size of a substation could vary form small (with one or two transformers and a dozen of switches) to extremely large (with several transformers and hundreds of switches plus other equipment). There are four different types of substations: the step-up transmission substation, the step-down transmission substation, distribution substations and the underground distribution substation. The step-up transmission substation receives the energy from the power generation plant and increases the voltage of the electricity to be transported more easily. The step-down transmission substation is the source for distribution lines and sub-

transmission lines. They are located at switching points on the power grid and it lowers the voltage to 69kV.

The distribution substations lower the voltage to 2.4kV in order to be used for the final consumer. Most of the distribution substations are located very close to final user and are commonly located inside the cities. In the other hand, the underground distribution substations are completely dedicated to lower the voltage of the sub-transmission and also it is located very close to the final users. For the substations there are many ways to be connected to the primary distribution system, but the most used are the radial configuration and the interconnected/complex configuration.

Some advantages of radial configuration networks is that it has low short circuit currents, simpler switching and less protecting equipment. The main disadvantage is that radial configuration is less reliable. For those reason the distribution systems are build and planned as a weak meshed network, and are operated as radial network (Taleski *et al;* 1997). Since radial configuration is simple to operate, economical, safe and can achieve a good reliability due to the high reliability of the components; the only problem with radial configuration is that when one component fails all the system fails. One when there is only one direct path form the high voltage feeder to the transformer and then to the load, we have a radial configuration. Most of the power distribution systems are operated as radial configuration networks but this tendency is changing (Baran and Wu, 1989).

For the complex configuration can be supplied by one or more sides, giving to it a great flexibility, allowing that certain components fail. A complex configuration is a made by the connection of many load-center secondary lines with circuit breakers. One advantage of this configuration is that it allows the removal of one piece, making easier the maintenance and repair of components.

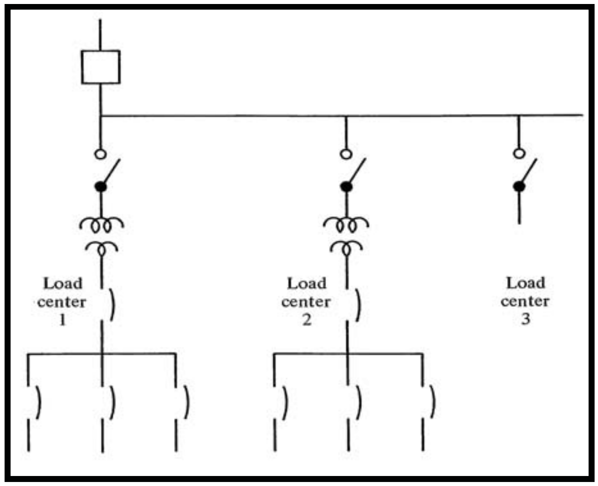


Figure 3-25: Radial Distribution Configuration

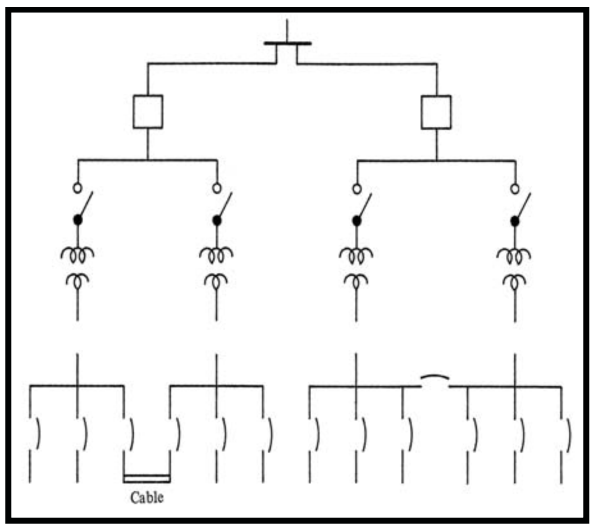


Figure 3-26: Complex Distribution Configuration

Source: Electric Power Transmission Systems, Eaton & Cohen, (1983)

3.3 Conclusion

The present chapter gave an explanation of what a component replacement problem is. An explanation of the origins and the importance of a component replacement analysis were give in detail. Also an explanation of how the power system gird work was given. The electric generation was explained in detail. A brief explanation of all the types of electric generation was given, including the

renewable and non-renewable sources of electric energy. Transmission network is explained in detail. The overhead, underground and submarine transmission lines were explained. Finally the distribution system was detailed also. Since the first half of the present research is based on the distribution system a detailed explanation was given. The primary goal of the power distribution system is to lower the voltage of the electricity arriving from the transmission lines and distribute it to the final costumer. The power distribution system is the area of interest because most of the energy interruption for the costumer happens for failure in the power distribution system. Finally the radial and the complex configuration were explained. The present research solves a replacement-scheduling problem for both configurations.

Chapter 4: Power Distribution System Schedule Optimization

4.1 INTRODUCTION

For the present research a Genetic Algorithm were developed. The main objective of the Genetic Algorithm is to find the best component replacement schedule for the power distribution system over a finite time horizon. Genetic Algorithms were used in the pas to solve similar problems for example, in 1997 Abdel-Magid *et al*; design a model for stabilization of a power generator. The authors took into account a range of different operating conditions and only one system stabilizer. This problem was solved by using Genetic Algorithms and an eigenvalue based objective function, where the main objective was to minimize the state vector. In 1999 Yangpin *et al;* determined the fault diagnostics at a nuclear plant by using probability and Genetic Algorithm. The last experiment was done in a 950 MW full size simulator, located at Beijing, China.

By 2003 Magid and Abido continued the work done by Abdel-Magid *et al*; by the formulation of the problem as a multi-objective function. The multi-objective function tried to optimize the whole system by taking into account the damping factor and the damping ratio. In this work a machine bus system was used to demonstrate the technique. Using this technique all the parameters are then considered adjustable. In 1996, Coit and Smith determined the optimal configuration of subsystems. This work demonstrates the reliability for the series-parallel systems. Again Genetic Algorithms were used to find the best solution for the given problem. The objective function of the given problem was to minimize the cost of the system and at the same time maximize the reliability of the system. In 2003 Genetic Algorithms was also used by Bris *et al;* along with Monte Carlo simulation in order to minimize the cost of the preventive maintenance of a series-parallel system. After, in 2005, Samrout *et al;* solved the same problem presented by Bris *et al*; using Ant Colony Optimization instead of a Genetic Algorithm. Samrout *et al;* compared them results with the results obtained by Bris *et al;* via cost and time.

Sundhararajan and Pahwa presented a new optimization method in 1994 that determines, for a radial distribution system, the best selection of capacitors to be placed. They proposed a Genetic Algorithm in order to find the objective function. The objective function was to minimize the power and energy losses in a radial distribution systems, and it takes into account as parameters of selection the size, type and location. Later in 2000, Levitin *et al*; used the model of Sundhararajan and Pahwa (1994) to find the optimal allocation of capacitors in the power distribution system, with the difference that this model also takes in consideration the different load parameters. Levitin *et al;* also used a Genetic Algorithm but using a integer encoding to represent the allocation of the different capacitors. In 2001 another capacitor replacement problem in a radial configuration distribution system was solve by Gallego *et al;* but they used two different methods to solve the problem. In one they used a Memetic Algorithm (with the local search being a Tabu Search) and a hybrid of Simulated Annealing with Tabu Search. There is many research in the area of the capacitor replacement problem. The other research's done in the same field are listed below:

- In 2004 using Ant Colony Optimization, Chiou *et al;*
- In 2004 using Genetic Algorithms, Masoum *et al;*
- In 2008 using Mixed-Integer Programming, Khodr *et al;*
- In 2008 using Ant Colony Optimization, Chang.
- In 2010 using Particle Swarm Optimization, Ziari *et al;*

The first half of the present research solves the component replacement-scheduling problem for the power distribution system for a finite planning horizon, with the objective of minimizing the total cost. The model was developed and applied for radial and complex configuration.

4.2 Formula Calculation.

The main objective of this part of the present research is to minimize the total cost of component

replacement schedule for the power distribution system. The objective function to minimize is the sum of the maintenance cost, unavailability cost and the purchase cost, and each parameter is dependent on the decision of KEEP or REPLACE the given component. To evaluate the current age of the component in the power grid the N.H.P.P (Non-Homogeneous Poisson Process) will be used. In the NHPP events occur at random over time and at an average rate of λ events per unit time. Moreover, from strategic and financial viewpoints, a firm's ability to analyze the elements of equipment degradation as well as addible market- and cost effects of replacements is crucial to competitive and continual technological advance markets (Chang, 2003). Hritonenko and Yatsenko in 2007 developed a continuous-time equipment replacement model under improving technology.

The rate at which events occur in a NHPP varies with time as determined by the intensity function, λ (t), which is an integral function of time (Arkin *et al.,* 2000). Since the properties to determine are commonly incomputable, abstraction has to be employed (Grund and Reineke, 2010). In the present work, the Crow/AMSAA (Army Material System Analysis Activity) model is used to determine the aging (increasing failure rates) for the different components in the power distribution system.

The failure intensity function for each component *l* in the system is given by,

$$E_l[N(\tau)] = \lambda_l \tau^{\beta_l} \tag{4.1}$$

$$\mu_l(\tau) = \lambda_l \beta_l \tau^{\beta_l - 1} \tag{4.2}$$

Where λ_l and β_l are NHPP parameters, and τ is the component age.

The expected number of failures by age τ on any one-year time interval of the component is calculated by:

$$\lambda_{i,j}(\tau) = \lambda_l[(\tau + 1)^{\beta_l} - \tau^{\beta_l}] \tag{4.3}$$

Maintenance cost of a component is the cost associated to maintaining the component in the power grid during its maintenance schedule. A good maintenance policy reduces a effective age of any component by a stated percentage of its actual age, but it does not affect the failure rate of the component. Basically, a maintenance optimization model is a mathematical model in which both costs and benefits of maintenance are quantified and in which an optimum balance between both is obtained (Dekker, 1996). The decisions are further complicated when capital costs and operation and maintenance (O&M) costs are decreasing at different rates, (Regnier *et al,* 2010). The maintenance cost of the component is calculated by using the following equation:

$$M_{i,j}(\tau) = \lambda_{i,j}(\tau). C_i \tag{4.4}$$

Where C_i is the cost of Repair for i^{th} component during j^{th} planning period.

The cost associated with the unavailability of electricity to the customers due to the network showdown during the maintenance of the system, is called the unavailability cost. The unavailability cost of the component is calculated by using the following equation:

$$U_{i,j}(\tau) = \lambda_{i,j}(\tau). r_{i,j}. I_t \tag{4.5}$$

Where I_t is the customer interruption cost and $r_{i,j}$ is the Repair Time of i^{th} component during j^{th} planning period.

For the complex configuration the cost of unavailability cannot be applied as directly as when is using a radial configuration. In radial configuration when a component fails the whole system fails, but for complex systems when one component fails not necessarily means that the whole system will fail. This particularity of the complex system makes the outage rate a nonlinear neither a separable model, such that the unavailability cost cannot be directly obtained as in the radial configuration. In order to calculate the unavailability cost, a separable function must be creates. This separable function will sum the contribution of each individual component to get the total unavailability cost. A Taylor series

expansion is used to estimate the distribution system unavailability. The equation used is a general formulation of the Taylor series expansion to approximate complex systems objective functions (Coit and Jin, 2001).

(4.6)

$$f(x) = f_0(x) + \sum_{i=1}^{n}(x_i - x_{0i})\frac{\partial}{\partial x_i}f(x)\big|_{\text{x-x0}} + \sum_{i=1}^{n}\frac{(x_i - x_{0i})^2}{2}\frac{\partial^2}{\partial {x_i}^2}f(x)\big|_{\text{x-x0}}$$
$$+ \sum_{i=1}^{n}\sum_{j>i}^{n}\frac{(x_i - x_{0i})(x_i - x_{0j})}{2}\frac{\partial^2}{\partial x_i \partial x_j}f(x)\big|_{\text{x-x0}} + \dots$$

For the complex system the opportunity costs are based on an approximation of the Taylor series expansion model as shown on equation 7.

(4.7)

$$U_{s,t} = U(u) \approx \sum_{i=1}^{n}(\mu_{l,t} - 0)\frac{\partial}{\partial u_{l,t}} + \cdots = \sum_{i=1}^{n}\left(\lambda_{l,t} r_l - 0\right)\frac{\partial \lambda_{l,t}}{\partial u_{l,t}}\frac{\partial U}{\partial u_{l,t}} = \sum_{i=1}^{n}\lambda_{l,t} r_l \frac{1}{r_l}\frac{\partial U}{\partial u_{l,t}}$$

By using the last equation the unavailability cost of each component in a complex system configuration can by estimated using the following equation:

(4.8)

$$U_{s,t} = \sum_{i=1}^{n}\lambda_{l,t}\frac{\partial U}{\partial u_{l,t}}$$

Then using equation 8 the total system unavailability can be separated and then approximate the contribution of each component. The individual contribution to the total system unavailability is determined by using equation 9.

(4.9)

$$\phi_{l,t} = \lambda_{l,t}\frac{\partial U}{\partial u_{l,t}}$$

The purchase cost is $P_{i,j}$, and it stands for purchase cost of a new i^{th} component during j^{th} planning period. Basically, the objective function with an N number of components and planning horizon K is formulated as following:

$$F_K = min \sum_{i=1}^{N} \sum_{j=1}^{K} [\, M_{i,j}(\tau) + U_{i,j}(\tau) + P_{i,j}] \tag{4.10}$$

Subject To, Budget Constraints.

4.3 Genetic Algorithm Developed

The genetic algorithm developed works with an binary encoding where 1 denotes to KEEP a component decision and 0 denotes to REPLACE a component decision. A 1 in a chromosome means that we are going to keep the component al least until the next planning period. This means that the particular component will incur total of Maintenance Cost and Unavailability Cost, and a zero purchase cost. Furthermore, when we encounter 0 in the chromosome then it's a replacement decision for that particular component and that incurs in a total of Purchase cost and also Unavailability Cost and zero maintenance cost because the component is new. At the beginning an initial population of 1000 chromosomes is created and run for a total of 50 generations.

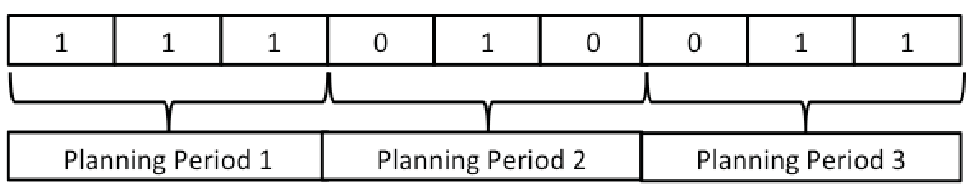


Figure 4-27: Chromosome Representation for the component replacement schedule problem

At each iteration all the chromosomes that form the initial population will be evaluated using the objecting function. Then for each particular chromosome the maintenance cost, the purchase cost and the unavailability cost are calculated. The selection factor chose for the Genetic Algorithm developed

was Elitism, where the 30% best solution will pass thru the genetic operators. For the crossover operator is done by getting all the none-genes of a parent chromosome join the even-genes of another parent chromosome to form a new “child” chromosome.

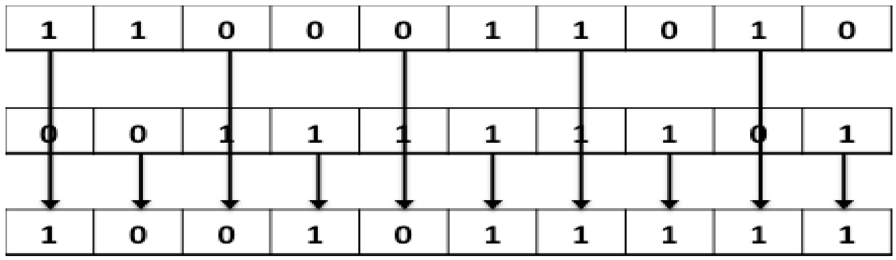


Figure 4-28: Figure operator of the Genetic Algorithm Developed

The same part of the population selected to be pass over the genetic operators undergo mutation. The mutation done in the present genetic algorithm selects at random one gene of the chromosome and changes its value for one different.

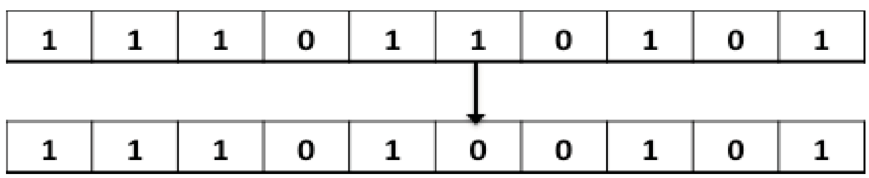


Figure 4-29: Crossover operator of the Genetic Algorithm Developed

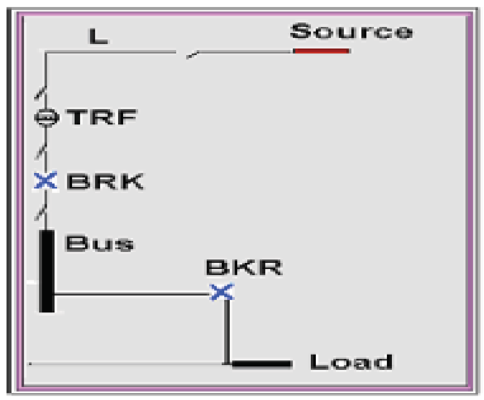
Then after the genetic algorithms created new chromosomes with the information of the chromosomes selected by elitism all of those new chromosomes are evaluated to know which are the best according to the objective function. Then selecting the 30% best fitted chromosomes creates a new population and the rest of the population is created at random again until the stopping criterion is reached.

4.4 NUMERICAL EXAMPLES

There will be two different numerical examples, the first one will be in the radial configuration, using 4 components and the second example will be using a complex configuration using eleven components.

4.4.1 Radial Configuration Example

A radial configuration leaves the station and passes through the network area with no normal connection to any other supply. This is a typical rural line with isolated areas.



Radial Configuration

Figure 4-30: Radial electricity distribution network

The first numerical example uses the radial configuration shown in the figure 4.3 and apply to it the genetic algorithm developed. The main objective is to find the best replacement schedule for the components of the radial configuration network. We will use the direct evaluation technique, in which we have a number of components ("N"), also the number of planning periods ("K"). An inflation rate of 1% is set to take car of the inflation for the budget schedule. The Genetic Algorithm evolves to find the chromosome with the minimum cost. In this example we are going to use four components: a Line 13.8kV, a Breaker 13.8kV, a Line of 600ft and finally a Switch. The table 4.1 summarizes the information used.

Table 4.1: Component data

Component	Asset initial age (τ_0)	λ_l (Outages/year)	$\beta_l\beta_l$	r_i (Hours/Outage)	C_i ($/Outage)	P_i ($)
1) Line 13.8kV	**20**	**1.9560**	**1.25**	**1.32**	**1500**	**45,000**
2) Breaker 13.8kV	**10**	**0.0036**	**1.60**	**83.12**	**1000**	**35,000**
3) Line 600 ft	**40**	**0.0055**	**1.80**	**26.51**	**1900**	**33,300**
4) Switch	**35**	**0.0061**	**1.85**	**5.60**	**700**	**10,000**

In the present example the replacement schedule was determined for N=4 components over the planning period of K=10.

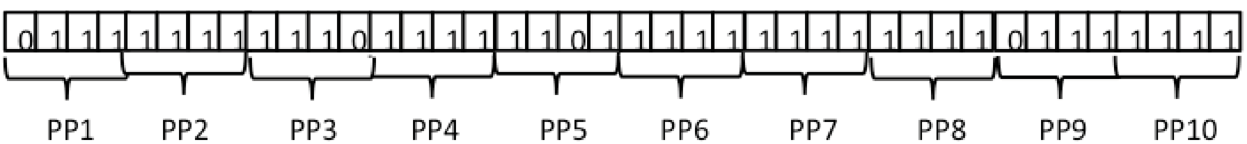


Figure 4-31: Recommended Chromosome for the Radial System

Figure 4.5 shows the best solution found by the genetic algorithm, where PP# denotes a planning period. Each PP# is divided in four cells the firs cell is for the line of 13.8kV, the second line for the breaker 13.8kV, the third Line 600ft, and finally the fourth the Switch.

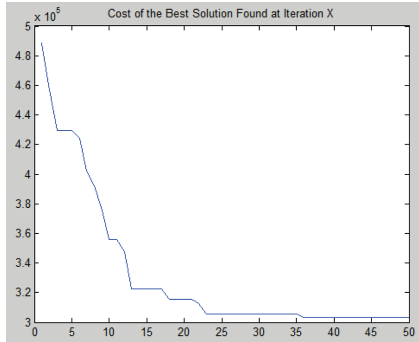


Figure 4-32: Graph showing the lowest total cost obtained at each interval

The graph in the Figure 4.6 shows the total cost of the fittest candidate obtained after evaluation of all the possible solutions at the end of each interval. The algorithm is run for 50 generations and the lowest total cost obtained at the end of 50th generation is $307,070.

4.4.2 Complex Configuration Example

A complex configuration is a series of interconnection of component that gives redundancy to the system distribution. The complex configurations are becoming each day common especially in urban areas such as cities for the high level of system reliability that they represent.

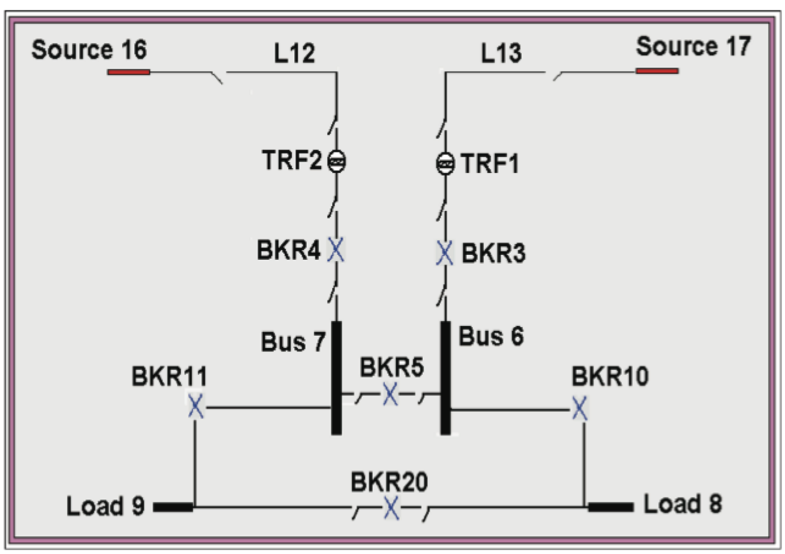


Figure 4-33:Complex Electricity Distribution Network

The second numerical example uses a complex configuration as is shown in figure 4.6 and the same genetic algorithm developed for the radial configuration is used for the complex configuration. The only difference between the Genetic Algorithm used before and the one used for the complex configuration is the calculation of the unavailability cost. The same parameters are used, but in this time there are 11 components instead of 4, but the planning period will remain the same. In this example the four components are: Components 12 and 13 representing supply lines, components 1 and 2 are the power transformers, buses are defined by components 6 and 7, and finally, breakers are component numbers 3, 4, 5, 10, 11 and 20. Breaker number 20 is normally open. Table 4.2 summarizes the information of the 11 components used.

Table 4-2: Complex system component data

Component	**Asset initial age (τ_0)**	λ_l **(Outages/year)**	$\beta_l \beta_l$	r_i **(Hours/Outage)**	C_i **($/Outage)**	P_i **($)**
1) Line 12	**35**	**1.8394**	**1.25**	**6.0177**	**5000**	**85,000**
2) Line 13	**38**	**2.2453**	**1.28**	**9.1432**	**5500**	**105,000**
3) Transformer 1	**28**	**0.2042**	**1.55**	**97.691**	**6000**	**120,000**
4) Transformer 2	**31**	**0.3478**	**1.75**	**103.69**	**7000**	**140,000**
5) Bus 6	**25**	**0.1982**	**1.73**	**5.9749**	**3000**	**95,000**
6) Bus 7	**29**	**0.2275**	**1.76**	**4.8964**	**3500**	**88,000**
7) Breaker 3	**31**	**0.1820**	**1.75**	**157.52**	**2000**	**75,000**
8) Breaker 4	**19**	**0.1560**	**1.72**	**132.94**	**1800**	**78,000**
9) Breaker 5	**32**	**0.0620**	**1.75**	**100.18**	**2500**	**76,000**
10) Breaker 10	**25**	**0.0058**	**1.52**	**43.8201**	**1800**	**56,000**
11) Breaker 11	**30**	**0.0066**	**1.59**	**48.1101**	**2000**	**48,000**

The first step to solve the problem is to find the minimal cut-sets for electric outage at either load 8 or load 9. The minimal cut-set are minimal combination of components that are not working that creates a system failure, in this cases an outage. The minimal cut-sets and the formula to calculate the outage rate, repair time and the unavailability for each cut-set are summarized on table 4.3.

Table 4-3: Outage rate and repair times of cut sets for outages at either load 8 or load 9

Cut set	**Outage rate**	λ_l **Repair time**	**Unavailability**
{BUS 7}	λ_7	r_7	$\lambda_7 r_7$
{BKR 11}	λ_{11}	r_{11}	$\lambda_{11} r_{11}$
{BUS 6}	λ_6	r_6	$\lambda_6 r_6$
{BKR 10}	λ_{10}	r_{10}	$\lambda_{10} r_{10}$
{LINE 13, TRF 2}	$\lambda_{13}\lambda_2(r_{13}+r_2)$	$\frac{(r_{13}r_2)}{r^{13}+r^2}$	$\lambda_{13}\lambda_2(r_{13}r_2)$
{TRF 2, BUS 6}	$\lambda_2\lambda_6(r_2+r_6)$	$\frac{(r_2r_6)}{r^2+r^6}$	$\lambda_2\lambda_6(r_2r_6)$
{BKR 4, BUS 6}	$\lambda_4\lambda_6(r_4+r_6)$	$\frac{(r_4r_6)}{r^4+r^6}$	$\lambda_4\lambda_6(r_4r_6)$
{LINE 12, LINE 13}	$\lambda_{12}\lambda_{13}(r_{12}+r_{13})$	$\frac{(r_2r_6)}{r^2+r^6}$	$\lambda_{12}\lambda_{13}(r_{12}r_{13})$
{BKR 3, BKR 4}	$\lambda_3\lambda_4(r_3+r_4)$	$\frac{(r_3r_4)}{r^3+r^4}$	$\lambda_3\lambda_4(r_3r_4)$
{TRF 2, BKR 5}	$\lambda_2\lambda_5(r_2+r_5)$	$\frac{(r_2r_5)}{r^2+r^5}$	$\lambda_2\lambda_5(r_2r_5)$
{LINE 12, BUS 6}	$\lambda_{12}\lambda_6(r_{12}+r_6)$	$\frac{(r_{12}r_6)}{r^{12}+r^6}$	$\lambda_{12}\lambda_6(r_{12}r_6)$
{TRF 1, LINE 12}	$\lambda_1\lambda_{12}(r_1+r_{12})$	$\frac{(r_1r_{12})}{r^1+r^{12}}$	$\lambda_1\lambda_{12}(r_1r_{12})$
{LINE 13, BUS 7}	$\lambda_{13}\lambda_7(r_{13}+r_7)$	$\frac{(r_{13}r_7)}{r^{13}+r^7}$	$\lambda_{13}\lambda_7(r_{13}r_7)$
{LINE 12, BKR 5}	$\lambda_{12}\lambda_5(r_{12}+r_5)$	$\frac{(r_{12}r_5)}{r^{12}+r^5}$	$\lambda_{12}\lambda_5(r_{12}r_5)$
{BKR 3, BUS 7}	$\lambda_3\lambda_7(r_3+r_7)$	$\frac{(r_3r_7)}{r^3+r^7}$	$\lambda_3\lambda_7(r_3r_7)$
{TRF 1, BKR 5}	$\lambda_1\lambda_5(r_1+r_5)$	$\frac{(r_1r_5)}{r^1+r^5}$	$\lambda_1\lambda_5(r_1r_5)$
{LINE 12, BKR 3}	$\lambda_{12}\lambda_3(r_{12}+r_3)$	$\frac{(r_{12}r_3)}{r^{12}+r^3}$	$\lambda_{12}\lambda_3(r_{12}r_3)$
{LINE 13, BKR 5}	$\lambda_{13}\lambda_5(r_{13}+r_5)$	$\frac{(r_{13}r_5)}{r^{13}+r^5}$	$\lambda_{13}\lambda_5(r_{13}r_5)$
{BKR 3, BKR 5}	$\lambda_3\lambda_5(r_3+r_5)$	$\frac{(r_3r_5)}{r^3+r^5}$	$\lambda_3\lambda_5(r_3r_5)$
{TRF 1, TRF 2}	$\lambda_1\lambda_2(r_1+r_2)$	$\frac{(r_1r_2)}{r^1+r^2}$	$\lambda_1\lambda_2(r_1r_2)$
{TRF 1, BUS 7}	$\lambda_1\lambda_7(r_1+r_7)$	$\frac{(r_1r_7)}{r^1+r^7}$	$\lambda_1\lambda_7(r_1r_7)$
{BKR 4, BKR 5}	$\lambda_4\lambda_5(r_4+r_5)$	$\frac{(r_4r_5)}{r^4+r^5}$	$\lambda_4\lambda_5(r_4r_5)$
{LINE 13, BKR 4}	$\lambda_{13}\lambda_4(r_{13}+r_4)$	$\frac{(r_{13}r_4)}{r^{13}+r^4}$	$\lambda_{13}\lambda_4(r_{13}r_4)$
{TRF 1, BKR 4}	$\lambda_1\lambda_4(r_1+r_4)$	$\frac{(r_1r_4)}{r^1+r^4}$	$\lambda_1\lambda_4(r_1r_4)$

{BKR 3, TRF 2}	$\lambda_3\lambda_2(r_3 + r_2)$	$\frac{(r_3 r_2)}{r^3 + r^2}$	$\lambda_3\lambda_2(r_3 r_2)$

Then the expected system unavailability at either loan 8 or load 9 is obtained by using the following equation:

$$f(U_{l8\cup l9}) = \lambda_7 r_7 + \lambda_{11} r_{11} + \lambda_6 r_6 + \lambda_{10} r_{10} + \lambda_{13}\lambda_2(r_{13}r_2) + \lambda_2\lambda_6(r_2 r_6) + \lambda_4\lambda_6(r_4 r_6) + \lambda_{12}\lambda_{13}(r_{12}r_{13}) + \lambda_3\lambda_4(r_3 r_4) + \lambda_2\lambda_5(r_2 r_5) + \lambda_{12}\lambda_6(r_{12}r_6) + \lambda_1\lambda_{12}(r_1 r_{12}) + \lambda_{13}\lambda_7(r_{13}r_7) + \lambda_{12}\lambda_5(r_{12}r_5) + \lambda_3\lambda_7(r_3 r_7) + \lambda_1\lambda_5(r_1 r_5) + \lambda_{12}\lambda_3(r_{12}r_3) + \lambda_{13}\lambda_5(r_{13}r_5) + \lambda_3\lambda_5(r_3 r_5) + \lambda_1\lambda_2(r_1 r_2) + \lambda_1\lambda_7(r_1 r_7) + \lambda_4\lambda_5(r_4 r_5) + \lambda_{13}\lambda_4(r_{13}r_4) + \lambda_1\lambda_4(r_1 r_4) + \lambda_3\lambda_2(r_3 r_2) \tag{4.11}$$

By using equation 11, the individual contribution for each component to the unavailability of the system ($\phi_{l,t}$) can be determined using the equations 4.12 thru 4.24

$$\phi_1 = \lambda_1[\lambda_{12}(r_1 r_{12}) + \lambda_5(r_1 r_5) + \lambda_2(r_1 r_2) + \lambda_7(r_1 r_7) + \lambda_4(r_1 r_4)] \tag{4.12}$$

$$\phi_2 = \lambda_2[\lambda_{13}(r_{13}r_2) + \lambda_6(r_2 r_6) + \lambda_5(r_2 r_5) + \lambda_1(r_1 r_2) + \lambda_3(r_3 r_2)] \tag{4.13}$$

$$\phi_3 = \lambda_3[\lambda_4(r_3 r_4) + \lambda_7(r_3 r_7) + \lambda_{12}(r_{12}r_3) + \lambda_5(r_3 r_5) + \lambda_2(r_3 r_2)] \tag{4.14}$$

$$\phi_4 = \lambda_4[\lambda_6(r_4 r_6) + \lambda_3(r_3 r_4) + \lambda_5(r_4 r_5) + \lambda_{13}(r_{13}r_4) + \lambda_1(r_1 r_4)] \tag{4.15}$$

$$\phi_5 = \lambda_5[\lambda_2(r_2 r_5) + \lambda_{12}(r_{12}r_5) + \lambda_1(r_1 r_5) + \lambda_{13}(r_{13}r_5) + \lambda_3(r_3 r_5) + \lambda_4(r_4 r_5)] \tag{4.16}$$

$$\phi_6 = \lambda_6[r_6 + \lambda_2(r_2 r_6) + \lambda_4(r_4 r_6) + \lambda_{12}(r_{12}r_6)] \tag{4.17}$$

(4.18)

$$\phi_7 = \lambda_7[r_7 + \lambda_{13}(r_{13}r_7) + \lambda_3(r_3r_7) + \lambda_1(r_1r_7)]$$

(4.19)

$$\phi_8 = 0$$

(4.20)

$$\phi_9 = 0$$

(4.21)

$$\phi_{10} = \lambda_{10}r_{10}$$

(4.22)

$$\phi_{11} = \lambda_{11}r_{11}$$

(4.23)

$$\phi_{12} = \lambda_{12}[\lambda_{13}(r_{12}r_{13}) + \lambda_6(r_{12}r_6) + \lambda_1(r_1r_{12}) + \lambda_5(r_{12}r_5) + \lambda_3(r_{12}r_3)]$$

(4.24)

$$\phi_{13} = \lambda_{13}[\lambda_2(r_{13}r_2) + \lambda_{12}(r_{12}r_{13}) + \lambda_7(r_{13}r_7) + \lambda_5(r_{13}r_5) + \lambda_4(r_{13}r_4)]$$

Then by using ϕ_1 the estimated contribution of component 1 to the total system unavailability can be obtained. In the present example the replacement schedule was determined for N=11 components over the planning period of K=10.

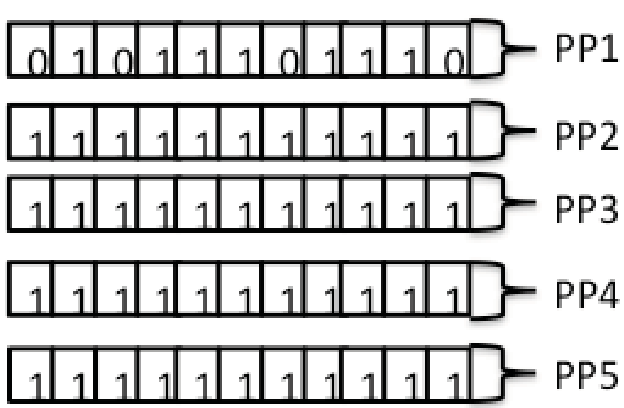


Figure 4-34: Recommended Chromosome for the Complex System

Figure 4.8 shows the best solution found by the genetic algorithm, where PP# denotes a planning period. Each PP# is divided in eleven cells, the chromosome has a total length of 55 genes but it is divided in the five planning periods in figure 4.8 for easy understanding.

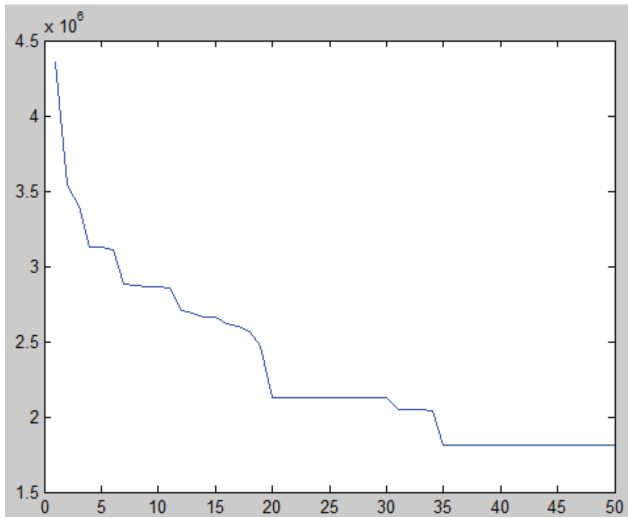


Figure 4-35: Graph showing the lowest total cost obtained at each interval

The graph in the Figure 4.9 shows the total cost of the fittest candidate obtained after evaluation of all the possible solutions at the end of each interval. The algorithm is run for 50 generations and the lowest total cost obtained at the end of 50th generation is $1,811,300.

4.5 CONCLUSION

A method is developed to determine the optimal component replacement schedule for the power distribution system in first half of the present research. The method developed was tested on a radial power distribution system configuration, which is the most common configuration used in the industry. The method developed was also tested on a complex power distribution system configuration, which is the power distribution configuration used most commonly in recent years. The model developed is Genetic Algorithm, were the user can specify the number of components used by the system and also the number of planning periods selected for the replacement schedule in consideration. This characteristic

makes the model more flexible so it can be use for any number of components and any number of planning periods. A generalized formulation was used in the Genetic Algorithm developed. The objective optimized by the Genetic Algorithm is the minimization of the total cost of the schedule selected. The total cost is the sum of the maintenance cost, unavailability cost and the purchase cost of a new component. The objective function was the same for both configuration (radial and complex) but there is a difference in the calculation of the unavailability cost. For the complex system it was calculated first the individual contribution of each component to total unavailability cost. Two examples were solved using the Genetic Algorithm developed giving very good component replacement schedules.

Chapter 5: Baggage Screening Problem

5.1 Introduction

In September of 2001, the United States suffered its biggest terrorist attack. The use of various planes as weapons to attack strategic places in the country changed the way we use to think about airport security. Also, in 1998 at Lockerbie, Scotland the Pan-American Flight 103 exploded, killing the 270 people on board. Both cases happened because terrorist bombs and/or terrorist passed through airport security. For this reasons US airport screening procedures have gone through significant changes to ensure passenger safety. In order to meet these improvements the Federal Aviation Administration (FAA) received funding from the Congress in the Omnibus Consolidated Appropriations Act of 1997. Over the four years prior to 2000, Congress provided the FAA with $1 billion for security; roughly one-third of this funding was for the purchase and deployment of security equipment at airports (Coughlin, *et al;* 2002).

Airport security was improved by the use of existing Explosive Detection Systems, Passenger-Baggage Matching and Automated Passenger Profiling and since the implementation of these measures; an important component to increase the security in airports was due to the improvements in baggage screening. The Transportation Security Administration (TSA) has focused in baggage screener training and the success of new procedures in identifying potential threats. Current screening procedures require passengers to arrive as early as three hours before the plane departure, exposing them to experience delays at screening points and in some extreme cases, airport closures. For example, in August 2005, the Portland International Airport was closed when two passengers breached the security checkpoint.

Surveillance equipment is the first aviation security operation in the United States. Aviation security began in 1970 in response to hijacking attempts. Since those attempts continue, beginning in December 1972, air carries were required to physically screen all passengers using metal detectors (National Research Council, 1996). In 1996, the Commission on Aviation Safety and Security

recommended to the aviation industry the use of existing explosive detection technologies, automated passenger prescreening, and positive passenger-baggage matching (McLay, *et all;* 2004). Some aviation security experts suggest that is more effective to give a more intense scrutiny of passengers perceived as greater security risks than increasing the screening intensity for all passengers, giving that 100% checked baggage screening is not cost-effective, and suggest that creating multiple levels of security for screening passengers may be more effective treating all passengers the same (Poole and Passantino; 2003). Some airports are implementing a prescreening security point in order to 100% check all the passenger baggage. The use of a prescreening system has operational implications in the screening of passengers with security screening devices and procedures.

Some airports use Computer Assisted Passenger Screening System (CAPSS) in order to inspect some potential terrorist passengers. CAPSS use passenger profile to indicate if a person is a potential terrorist and needs to pass through further inspection (O'Harrow, 2002). Some research point out that implementing CAPSS is not enough to warranty the security of an airport. Any airport that uses CAPSS to select passengers for increased scrutiny is bound to be less securing than other airport that uses systems that randomly select passengers for thorough inspection (Chakrabarti and Strauss; 2002). Chakrabarti and Strauss presented an algorithm called Carnival Booth that demonstrates how a terrorist can defeat the CAPSS. Using a combination of statistical analysis and computer simulation they evaluated the efficacy of Carnival Booth and illustrate that CAPSS is an ineffective security measure. Perisco and Petra analyzed the implementation of the Computer-Assisted Passenger Pre-Screening System (CAPPS II), which is the tool to select passengers for screening. CAPPS confirms passenger's identities, performs criminal identity and checks credit. Also receives additional information, such as residence, home ownership, income, and patterns of travel and purchases used to construct a predicted threat rating, finding that even though the CAPPS system does perform well that does not improve the 24-percent error in the baggage screening. In 2008 Sahin and Feng did a research on passenger

information incorporated into a two-level checked-baggage screening system to determine the screening strategy for different subsets of passengers.

In 2002 Frederickson and LaPorte applied the concepts and logic of high–reliability organizations to airport security operations in their study. The contemporary decision theory is built on the logic of limited or buffered ration ability and is based on the study of error–tolerant organizations, the concept of high–reliability organizations is based on the study of nearly error–free operations. In 2011, Weiss developed along with MITRE Corporation the Dynamic Airport Security Model. The model, a fast-time desktop simulation, accepts the airport layout, security procedures and threat vectors (path-weapon combinations) as inputs and models the performance of the airport's defense against those threat vectors. Moreover, Xiaofeng *et al.,* considered a model for an airport security system in which the declaration of a threat is based on the joint responses of inspection devices. They obtained the false alarm and false clear probabilities, and compared the response system with two other independently operated systems. They modeled group passengers in a manner that minimizes the false alarm probability while maintaining the false clear probability. Kwang and Youn studied the failures concerned with detecting prohibited items form air passenger or carry-on luggage, where there are three major factors affecting the effectiveness of passenger screening: human resources, equipment, facilities, procedures and responsibility structures. In their study they utilized an AHP (Analytic Hierarchy Process) analysis on surveyed data about the relative importance of the factors and elements concerned with the improvement of passenger screening.

Jacobson and Karnani in 2005, proposed a different integer-programming model for obtaining optimal baggage screening security device deployment. The resulting model was designed to optimally determine where to assign the different baggage screening security devices and which baggage screen, where optimality is based on minimizing either the number of uncovered baggage segments, the number of uncovered flight segments, or the number of uncovered passenger segments. Later, Lee and Jacobson

evaluated the effectiveness of sequential aviation security screening policies using a conditional probability inequality to develop an upper bound and attaining the set of optimal assignments for a given realization of passenger risk. The sequential passenger assignment problem has been formulated as a stochastic process, with the objective of maximizing the overall true alarm rate, subject to device capacity constraints.

In terms of what should be reported by a screening system and its performance characteristics, certain guidelines have been set by aviation authority (Singh and Singh; 2003). Most of the devices that are today in use are Explosive Detection Systems (EDS). For some items, the EDS machine cannot discern a difference between common products and known threat items. With a projected rate of 1.5 billion checked bags per year, and a rejection rate of 30 percent means 450 million bags per year—more than 1.2 million per day. All these bags need either further screening by another technology or hand search. In each case, the additional machine, time, or labor requirements for more-intensive additional screening of more than a million bags per day are very onerous (Butler and Poole; 2002). In 2007, Feng assessed the risk and cost effectiveness of various baggage-screening systems by investigating the setting of specifications or thresholds on security responses.

A Security Equipment Integrated Product Team (SEIPT) was formed within the FAA to acquire and deploy security equipment using the $144 million appropriated by Congress (Rao and Dickey, 1999), with the goal of making nearly impossible for a bomb to enter an aircraft. The FAA has developed several software packages to help in the determination of which baggage screening devices to buy, which airports need more those devices, how many devices are located in each airport and where the devices are located in the airport. One of the most important software that the FAA has developed is the Checked Baggage Screening (CBS) Model. The Checked Baggage Screening Model is an operational cost model developed by the FAA to be used by the FAA personnel, airline analysts and some economists.

5.2 Baggage Screening Problem

Different guidelines have been set by aviation authority to specify what should be reported by a screening system and its performance characteristics (Singh and Singh, 2003). Most baggage screening devices that are in use today at airports are Explosive Detection Systems (EDS). In some instances, the EDS machine cannot detect the difference between common products and known threat items. For example, with an estimated rate of 1.5 billion checked bags per year, and a rejection rate of 30 percent, then, more than 1.2 million bags will be rejected per day. All these rejected bags need either further screening by another technology or by hand search. In each case, the additional machine, time, or labor requirements for more-intensive additional screening are very onerous (Butler and Poole, 2002).

A Security Equipment Integrated Product Team (SEIPT) was formed within the FAA to acquire and deploy security equipment using the $144 million appropriated by Congress (Rao and Dickey, 1999), with the goal of making it nearly impossible for a bomb to enter an aircraft. The FAA has developed several software packages to help determine which baggage screening devices to procure, select which airports need more of those devices, how many devices will be located in each airport and where the devices will be deployed within the airport. One of the most important software packages that the FAA has developed is the Checked Baggage Screening (CBS) Model. The CBS Model is an operational cost model developed by the FAA that helps forecast the costs of implementing some different baggage screening strategies. These strategies are based on different (one to ten) levels of security screening devices where at each level a specific device checks the bags; this system is shown in Figure 5.1.

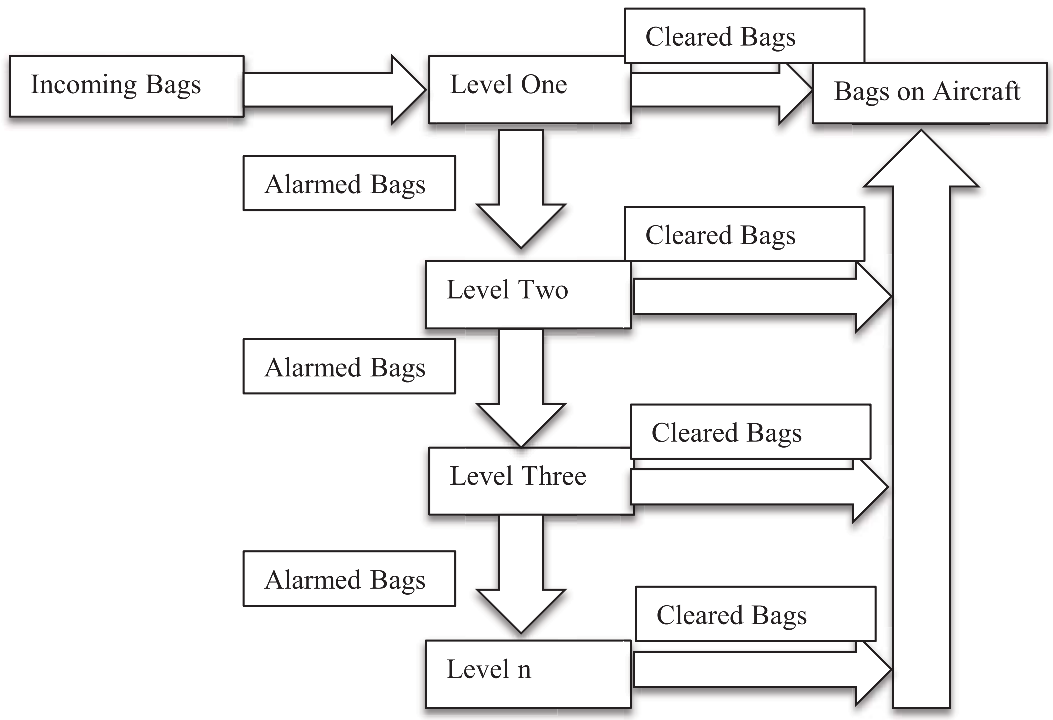


Figure 5-36: Baggage Screening System

On each level one type of device could be selected, for example if on level one the device selected is of type X, only type X devices could be located at that specific level. The number of bags expected to enter at that level and the throughput of the device selected at that specific level determines the number of devices per level. One advantage of the CBS model is that it can be used to determine different strategies for different airports. The CBS model gives the expected annual cost for purchase and operating the devices. The strategy is based on the number of flights that arrives or departs from a specific airport. Also, CBS projects the expected throughput of each baggage-screening device. The expected direct cost per expected prevented attack to the expected cost of an aviation terrorist incident provides one measure for the cost effectiveness of 100% checked bag screening (Jacobson and Karnani; 2005).

A bag is alarmed when after screening a bag the security device finds something unfamiliar on it. A bag is cleared when after screening a bag the security device does not find anything unfamiliar on it.

Each bag has the possibility of containing an explosive (called threated bags), and also each device has the probability to fail detecting a threated bag. Those probabilities lead to four different outcomes from a device, as shown in Figure 5.2

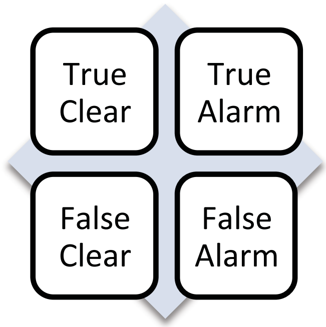


Figure 5-37: Security Device Findings

A true clear is when a security device successfully checks a bag and finds that it is safe from any threat; where a false clear is the opposite, a false clear is when a security device finds that a bag does not has any threat, when actually it contains a threat. A true alarm is when a security device successfully finds a threated bag, and a false alarm is when a security device finds a threat on a bag when actually the bag does not contain any threat. Each outcome has an associated cost, for example when a bag is said to be alarmed, there is an extra effort on it (time and resources) to inspect it and know if it really contains a threat. An alarmed bag needs to be inspected by a law enforcement officer or a bomb squad; also alarmed bags could represent a problem because the concourse or the airport must to be evacuated. A false clear also represent costs, such as the property cost, but most important the cost of the loss of lives.

There is no prefect baggage-screening device; each one has the probability of false clear or false alarm. At each level of the strategy chosen, there is the probability to a bag be alarmed and be sent to the next level to be checked again or be cleared to go directly to the aircraft. The main objective of this research is to find the best strategy of baggage screening devices that minimizes the total cost, where the final cost includes the purchase cost, the maintenance cost and also the false alarm and false clear costs.

In this research the CBS model cost function is used, also 40 different baggage-screening devices are considered. There is set that the maximum number of levels are 10, giving a total of $8.15 * 10^{48}$ possible strategies. Since it is almost impossible for any computer to solve such number of strategies in a respectable amount of time, the use of metaheuristics is proposed. In 2004 Candalino *et all*, solve a similar problem using simulated annealing, with positive results. Metaheuristics are a class of approximate methods designed to solve hard combinatorial optimization problems arising within various different areas (Crispim and Brandao; 2005).

The second half of the present research broadness the cost function of the baggage screening problem and present the best methodology for selection of Explosive Detection Systems which will minimize the total cost. 40 different devices will be used as the Explosive Detection Systems available for the current problem. A decision will be chosen between 2 up to 10 levels of security, giving $1.05*10^{16}$ possible combinations of devices. For the difficulty of the task of obtain a optimal solution among that number of possible solutions, two metaheuristic method will be used to solve the problem. There are several Metaheuristics methods such as genetic algorithms, ant colony, monkey algorithms or memetic algorithms. The second half of this research will concentrate in the use of memetic algorithms and genetic algorithm.

Chapter 6: Optimal Aviation Baggage Screening Strategy

6.1 Formula Calculation

The expected number of bags screened per year (S) is shown in equation 6.1, where S is equal to the estimated maximum number of bags arriving per hour over a ten year period considered in this study.

$$S = \left(\frac{Maximum\ bags\ per\ hour}{10}\right) * 24 * 365 \tag{6.1}$$

The expected annual cost of false clears and false alarms are shown in equations 6.2 and 6.3, respectively

$$FCC = C_{FC} * S * r_T * \left[P_{FC}(\mathbf{1}) + \sum_{\boldsymbol{i}=\mathbf{2}}^{\boldsymbol{n}} \boldsymbol{P_{FC}}(\boldsymbol{i}) \prod_{\boldsymbol{j}=\mathbf{1}}^{\boldsymbol{i}-\mathbf{1}} \left(\mathbf{1} - P_{FC}(\boldsymbol{j})\right)\right] \tag{6.2}$$

$$FAC = C_{FA} * \boldsymbol{S} * (\mathbf{1} - \boldsymbol{r_T}) * \prod_{\boldsymbol{i}=\mathbf{1}}^{\boldsymbol{n}} P_{FA}\,(\boldsymbol{i}) \tag{6.3}$$

In equation 6.2, the False clear cost is composed of the cost of a false clear times the expected number of bags screened per year times the probability of a threat, times the final probability of a false clear after the bag has passed through all the levels. In equation 6.3, the expected annual cost of a false alarm is comprised of the cost of a false alarm times the expected number of bags screened per year times the probability of a bag non-containing a threat times the probability of a false alarm after passing all the levels.

The number of bags per level i is found by multiplying the maximum number of bags per hour times the probability of a false alarm up to level $i-1$ plus the probability of a true alarm up to level $i-1$, as shown in Equation 6.4, and the number of devices required per level i can be calculated using Equation 6.5.

(6.4)

$$Bags(i) = Max. number\ of\ bags\ per\ hour * \left\{ \left[(1 - r_T) * \prod_{j=1}^{i-1} P_{FA} \right] + \left[r_T * \prod_{j=1}^{I-1} \left(1 - P_{FC}(j)\right) \right] \right\}$$

(6.5)

$$evices(i) = Bags(i)/Throughput(device_i)$$

The total cost at level i is composed of the number of devices at level i times the purchase cost of device used at level i as is shown in Equation 6.6 and the total purchase cost is shown in Equation 6.7.

(6.6)

$$Level\ Cost\ (i) = Devices(i) * PurchaseCost(device_i)$$

(6.7)

$$TPC = \sum_{i=1,2,\ldots,n} Level\ Cost(i)$$

Finally, the Annual Total Annual Cost is shown in Equation 8 and it is comprised of the annual purchase cost, the operating cost, the false clear cost and the cost of a false alarm over a ten-year period.

(6.8)

$$TAC = (TPC/10) + C_O \sum_{i=1}^{n} Devices(i) + FCC + FAC$$

6.2 Genetic and Memetic Algorithm Developed

The genetic algorithm developed works with a permutation encoding. Figure 6.1 is the graphic representation of the chromosome used. Each cell of the chromosome has a number, which represent the machine used. Also each cell is in order where the first cell represents the first level and the second cell represent the second level and so on.

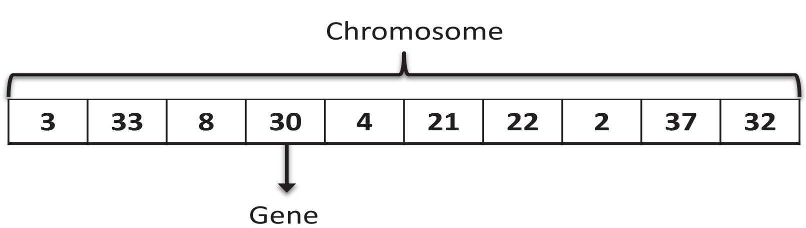


Figure 6-38: Chromosome Representation

Some of the characteristics of the Memetic Algorithms and Genetic Algorithms are that: MA’s and GA’s work with a coding of the parameter set, not the parameters themselves, MA’s and GA’s search from a population of points, not a single point, MA’s and GA’s use payoff (objective function) information. MA’s and GA’s use probabilistic transition rules, not deterministic rules, but only MA’s search local solutions to improve the fitness of the current population. A Memetic Algorithm is a hybrid between a local search and a Genetic Algorithm. The Memetic Algorithm uses the same evolutionary operators than the Genetic Algorithm (crossover and genetic operator). The crossover operator combines the information of two chromosomes (called parent chromosomes) in order to obtain a new chromosome that contains the information of both parent chromosomes. The crossover operator tries to get a better solution by combining two already good solutions, as in the nature when a child obtain the best from both parents. An example of a crossover operator is shown in figure 6.2.

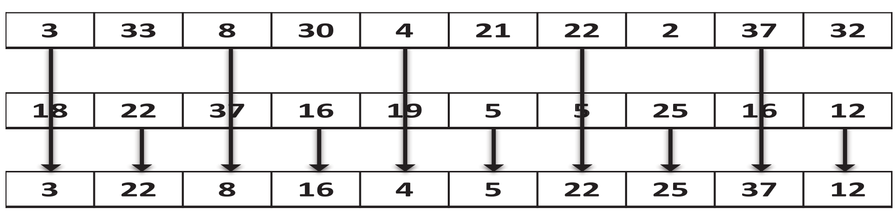


Figure 6-39: Crossover operator for the Baggage Screening Problem

.

The mutation operator uses the information of a chromosome and modifies it by changing the information of one cell of a chromosome for another completely different. Mutation is a genetic operator that alters one or more gene values in a chromosome from its initial state. This can result in an entirely new gene values being added to the gene pool. With these new gene values, the memetic and genetic algorithms may be able find a better solution. Mutation is an important part of the genetic search as it helps to prevent the population from staying at any local optima. An example of a mutation operator is shown in figure 6.3.

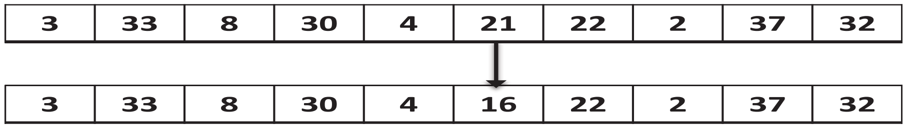


Figure 6-40: Genetic Operator Baggage Screening Problem

Memetic Algorithms are different from Genetic Algorithm for the use of a local search. Local search algorithms move from solution to solution in the space of candidate solutions (the *search space*) by applying local changes, until a solution deemed optimal is found or a time bound is elapsed. Examples of local search algorithms are WalkSAT, the 2-opt algorithm for the Traveling Salesman Problem, and Tabu Search. One of the most widely used and effective local search methods is the Tabu Search.

One of the advantages of Tabu Search is that it can be superimposed on other procedures to prevent them from becoming trapped at locally optimal solutions. Tabu Search is extremely popular for solving large combinatorial problems in many different areas. It is been used to solve a wide range of hard optimization problems such as job shop scheduling, graph coloring (related), the Travelling Salesman Problem (TSP) and the capacitated arc routing problem. A distinguishing feature of Tabu Search is its exploitation of adaptive forms of memory, which equips it to penetrate complexities that often confound alternative approaches (Glover and Laguna, 1999). Tabu Search begins with the initialization of a Tabu List. A local search is then used to scan the neighbors from the initial solution. From the solutions found among the neighbors the best is chosen and set as the new initial solution. The most common way to do the local search is to switch to genes of the chromosome as it is shown in figure 6.4.

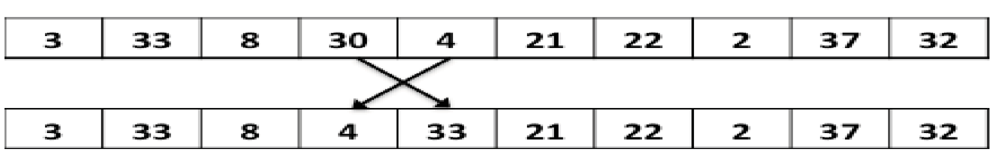


Figure 6-41: Local Search for the Memetic Algorithm

The particular aspect of the Tabu Search is that the switch move that gives the best solution is placed on the Tabu List. The entire moves that are in the Tabu List are penalty moves that prevent the repetition of the same solutions over time. However, an aspiration criterion can be specified such that a prohibitive move can still be admissible if this criterion is satisfied. The iterative process of the memory enhanced local search is repeated until a termination criterion is met. The best solution found during a TS run is regarded as the optimal solution.

The Memetic Algorithm used in this research first finds a random population of 100 chromosomes that will be evaluated. Once, the fitness value of the initial random population is found, the chromosomes will be sorted by it fitness value, from the more fitness value to the lowest fitness value. After the chromosomes are sort the 40 best solutions are set apart to give an initial population to the local search. In this paper we take the idea of a Tabu Search of switching the values of the genes in the chromosome to generate new solutions. The criterion used is to switch all the pair genes from the 40 best solutions to generate new neighbor solutions. Then fitness value is obtained of the new set of chromosomes and sorts them again, and the best 40 solutions are keep to be used for the genetic operators.

The crossover operator used in is set to choose one gene from parent 1, then one gene from parent 2, then a gene from parent 1 and so on until the length of the gene is met. The reason to doing this type of cross over operator is because it gives more information from both parents and also because it gives a son chromosome that have a variety of information from both parents. Also another child is born with the information of the same pair of parents both in this case the gene 1 will be the information of the gene 1 at parent 2 instead of the information of gene 1 in parent one, then the gene 2 of the son will be the information of the gene 2 of the parent one and so on until the length of the chromosome is meet. After getting all the new chromosomes all of those are set apart to be join with the chromosomes generated by the mutation operator.

The mutation operator used is set to be 1% and since the length of the chromosome used in this paper is set to be 10, the maximum information changed is set to be one gene. That gene that is going to be changed is selected at random and also the information of that gene is changed at random. All the new chromosomes generated by doing the mutation operator were joining with the chromosomes found using the crossover operator and for all of these new chromosomes the fitness value is obtained. Finally this values are compared with the values obtained after the local search and the best 100 solutions are then set to be our new initial population, this process is repeated 1000 times. To solve the baggage screening problem two different programs were developed using Matlab®, Figures 6.5 and 6.6 show the pseudo code of the developed algorithms.

```
Begin;
        Generate random population of P solutions (chromosomes);
        For i = 1to number of generations
                For each individual i ∈ P:
                        Calculate fitness (i);
                          Sort individuals from highest fitness value to lowest fitness value;
                    End;
                End;
        For best 40 solution = P2;
        For each individual i ∈ P2;
                Do Crossover (i);
                        Select two parents in order from P2, i_a and i_b;
                        Generate on offspring i_c and i_d= crossover (i_a and i_b);
                        i_c and i_d go to P3;
                End;
                Do Mutation (i);
                        For each individual i ∈ P2:
                        Generate on offspring i_2= mutation (i);
                        i_2 go to P3;
                End;
        End;
        For each individual i ∈ P3;
                Calculate fitness (i);
                   Sort individuals from highest fitness value to lowest fitness value;
         End;
        Best 100 P3 = P;
        Check if termination=true;
End;
```

Figure 6-42: Genetic Algorithm pseudo code

```
Begin;
        Generate random population of P solutions (chromosomes);
        For i = 1 to number of generations
                For each individual i ∈ P:
                        Calculate fitness (i);
                                Sort individuals from highest fitness value to lowest fitness value;
                        End;
                End;
        For best 40 solution = P2;
        For each individual i ∈ P2:
                Do local-search (i);
                        Calculate fitness of the offspring solutions;
                End;
        End;
        For best 40 solution = P3;
        For best 40 solution = P4;
        For each individual i ∈ P3;
                Do Crossover (i);
                        Select two parents in order from P3, i_a and i_b;
                        Generate on offspring i_c and i_d= crossover (i_a and i_b);
                        i_c and i_d go to P4;
                End;
                Do Mutation (i);
                        For each individual i ∈ P3:
                        Generate on offspring i_2= mutation (i);
                        i_2 go to P4;
                End;
        End;
        For each individual i ∈ P4;
                Calculate fitness (i);
                        Sort individuals from highest fitness value to lowest fitness value;
        End;
        Best 100 P4 = P;
        Check if termination=true;
End;
```

Figure 6-43: Memetic Algorithm pseudo code

6.3 NUMERICAL EXAMPLE

In the present research, the CBS model cost function is used with 40 different baggage-screening devices. The maximum number of possible levels for screening is 10, giving a total of $8.15 * 10^{48}$ possible strategies; the data used was obtained from previous studies (Virta *et al*., 2003; Candalino jr. *et al*., 2004). Table 6.1, shows the information for the different devices considered. Each baggage-screening device has a specific False Alarm probability (P_{FA}), False Clear probability (P_{FC}), purchase cost and throughput (bags per hour).

Table 6-1: Data Summarized

Device #	Cost	(P_{FA})	(P_{FC})	Throughput
1	900000	0.4	0.075	1200
2	330000	0.25	0.095	600
3	250000	0.2	0.085	100
4	850000	0.15	0.075	100
5	850000	0.3	0.065	254
6	3000	0.5	0.055	1000
7	2000	0.1	0.0545	2
8	500	0.5	0.0535	150
9	30000	0	0.0825	10
10	5000	0.5	0.075	1000
11	965000	0.25	0.075	50
12	850000	0.15	0.095	251
13	965000	0.25	0.085	260
14	200000	0.2	0.075	62
15	80000	0.25	0.065	80
16	70000	0.15	0.0755	80
17	45000	0.15	0.0845	66
18	90000	0.5	0.0835	1000
19	560000	0.2	0.0525	24
20	80000	0.4	0.075	600
21	800000	0.15	0.05	50
22	50	0.2	0.095	1
23	83000	0.2	0.085	180
24	16600	0.3	0.075	360
25	1500000	0	0.065	12
26	965000	0.15	0.0755	120
27	850000	0.15	0.0945	100
28	330000	0.18	0.0435	20
29	450000	0.25	0.0825	100
30	2000000	0.25	0.053	62
31	80000	0.12	0.095	80
32	57000	0.1	0.095	57
33	57000	0.25	0.085	80
34	70000	0.15	0.075	80
35	45000	0.15	0.0965	66
36	200000	0	0.055	1
37	600000	0.2	0.06	80
38	60000	0.5	0.0635	1000
39	200	0.1	0.0825	12
40	450000	0.25	0.075	550

For each bag there is a probability that it contains a threat, in this case the probability of a threat (r_T) is set to be $5.005 * 10^{-10}$. The annual operating cost per device (C_O) is $125,000.00, the cost of a false alarm (C_{FA}) is set to be $30, and finally the cost of a false clear (C_{FC}) is $1,400,000.00. The maximum number of bags per hour set in this research is five thousand. The expected number of bags screened per year (S) is shown in equation 1, where S is equal to the maximum number of bags arrived per hour over the ten years that are going to be analyzed times the 24 hours of a day times the 365 days of a year. The Matlab Program developed used the pseudo code of above and the information previously stated. It was run in Matlab R2010a by Matworks® on a 2.7 GHz Intel Core i7m, 4GB 1333MHz DDR3 computer. The solutions obtained are summarized in the table 6.2 and 6.3.

Table 6-2: Genetic Algorithm Final Solutions

Solution GA	Cost GA
2 12	7.1684e+007
2 2 12	3.1336e+007
2 2 2 31	2.1356e+007
6 2 2 2 31	1.7084e+007
6 6 2 2 2 12	1.5184e+007
6 6 6 2 2 2 32	1.4165e+007
6 6 6 6 1 2 2 31	1.3817e+007
6 6 6 6 2 1 24 23 32	1.3773e+007
6 6 6 2 2 2 23 23 8 7	1.4192e+007

Table 6.3: Memetic Algorithm final solutions

Solution MA	Cost MA
2 12	7.1684e+007
2 2 12	3.1336e+007
2 2 2 31	2.1356e+007
6 2 2 2 31	1.7084e+007
6 1 1 2 2 12	1.5425e+007
6 6 6 2 2 2 32	1.4165e+007
6 6 6 6 1 2 2 31	1.3817e+007
6 6 6 6 6 2 2 23 32	1.3618e+007
6 6 6 6 6 6 6 2 23 32	1.3522e+007

Both programs were run in the same computer, for the Genetic Algorithm it took almost a minute to give the results for the nine different levels. In the other hand the Memetic Algorithm run for 3 minutes to get the solutions for the nine levels. In both cases in all the solutions the device number 2 were used and from the level 5 also device number 6 were used in all the solutions. It is assumed that both devices has some advantages over the other devices because it gives a better strategy to use the same machine at different levels than only use it at one level. Device number 2 has a throughput of 600 bags per hour and a cost of $330,000.00 per machine, making it in balance between cost and number of bags screened per hour. Also, machine #2 has a False Alarm probability of 25% and False Clear probability of 9.5%, where even though those probabilities are not the lower they are low enough to have low cost for false alarms and false clears. Also device #6 has the same advantages as device #2 with a throughput of 1000, which is more than device 2, and a cost of $3000.00. Device #6 is set in the first levels because it has a big output of bags per hour at a low cost, and also with a low False Clear probability (5.5%), even though the probability of a false alarm is high (50%), the low purchase cost and high output lower the cost of the solution in the maximum levels.

As shown in table 1 the best solution found is at level 10 using the following distribution of devices: 6 6 6 6 6 6 6 2 23 32, at a cost of 1.3522e+007 dollars. The solution says that

in order to reduce costs at minimum from level 1 to level 7 use device #6, in level 8 use device #2, for level 9 use machine #23 and finally for level 10 use machine #32. This solution implies that for level one use 50 devices, at level two 26 devices, at level three 13 devices, at level four 7 devices, at level five 4 devices, at level seven 2 devices, and finally at level seven 1 which means that a total of 102 of machines #6 need to be buy. It is important to let know that this layout needs to be followed because it is not the same to have the specific number of machines in each level than having all the 100 machines in the same level due to the restriction that when a bag is alarmed need to go to the next level. For level 8 through level 10 only use one device at each level. The best solution layout is summarized at Figure 6.7.

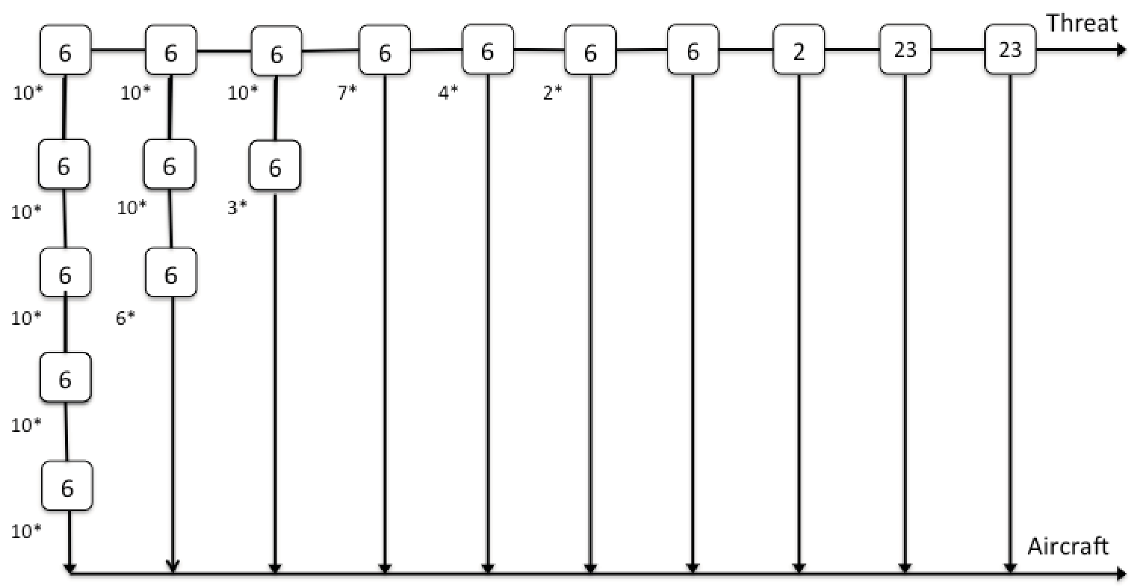


Figure 6-44: Best arrangement of machines found

Both Methaheuristics perform well obtaining solutions. At the beginning the genetic and the memetic algorithm gave exactly the same best solution, but at level nine and level ten the Methaheuristics found different best solutions. The memetic algorithm performs better finding an optimal solution, but there is an important difference in time while finding the solution. Table 6.4 shows the performance of the genetic algorithm and memetic algorithm at the level eight where both Methaheuristics found the same solution.

Table 6.4: Graphs showing the lowest total cost obtained at each interval

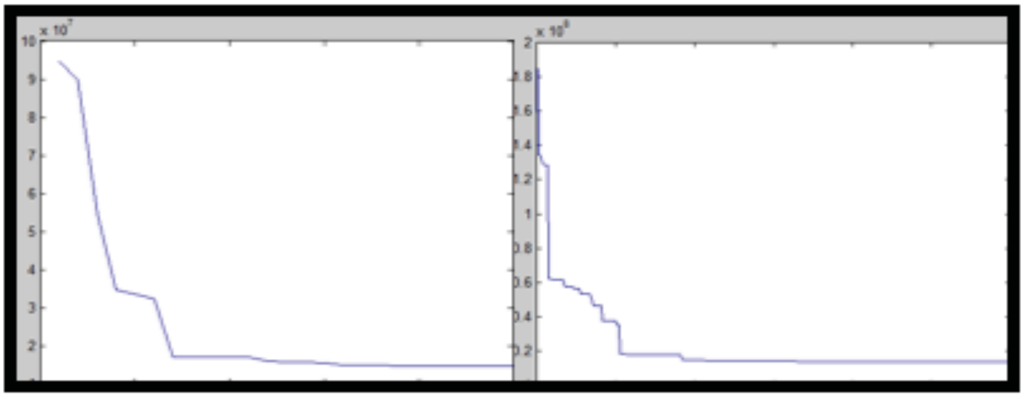


Table 6.4 shows the performance of both Methaheuristics when they found an optimal solution, the first table is the performance of the Genetic algorithm and the second table is the performance of the memetic algorithm. It can be seen that the memetic algorithm finds in less iteration the best solution.

6.3.1 Sensitivity Analysis

A sensitivity analysis were performed in order to find what is the most important parameter in order to choose an optimal solution, the sensitivity analysis was only performed using the memetic algorithm, since it found the best solution. The fist parameter to be changed was the cost of a false clear, because it seems to be the highest cost in the function. By setting a value of 0 to a false clear cost the best solution also changes from the number of levels and by the number and by the total cost. The best solution without a false clear cost is 6 6 8 8 39 39 9, at a total cost of $1.6413e+005. Setting a false clear in zero means that it is really not important to detect the threats at all, which makes no sense, that's why the optimal solution found changes drastically in cost and in number of levels needed, because since it is not important to really detect the threats the program only found the cheapest solution that gives less false alarms. Another parameter changed was the false alarm cost, it is also set to be zero, and the best solution found to be 6 6 at a cost of $9.6622e+006 and as the levels were increasing the cost is also increasing. This means that the cost of a false alarm is directly correlated with the number of levels needed in the strategy.

Other parameter set to zero was the operating cost, by doing this we are giving the freedom to choose more devices per level since the cost of using them does not really matter, as an observation makins the false clear and the operating cost to zero gives similar solution at similar optimum costs. The best solution found when the operating cost was set to be zero is 6 6 39 39 39 7 7 at a total cost of $1.6841e+005. Finally the most important parameter changed was the probability of a threat, since the other parameters are almost impossible to change, but the probability of a threat is the only one that really can constantly change. At the beginning the probability of a threat was set to be 5.005*10^-10, so it was changed to be 5.005*10^-8 and 5.005*10^-12, in other words a little bit more probable and a little bit less probable to a threat to happen. When the probability of a threat is 5.005*10^-8, so that is more probable to threat to happen, the solution is 6 6 6 6 6 40 40 21 increasing the cost of the optimal solution to $2.5883e+007, while in the other hand lowering the probability of the threat to 5.005*10^-12 gives an optimal solution with nine levels: 6 6 6 6 6 2 2 23 32 at a total cost of $1.3470e+007. Also, if the population size is change to be 1000 instead of a 100 the Matlab program to run for almost 15 minutes but the same solution was reach, so that we can assure that the optimal solution were found. If the population is changed to be 10 instead of 100 the program run faster, it finished searching for a solution in almost 20 seconds but the solution found is not the optimal solution found when the population size were 100 chromosomes.

6.4 Conclusion

A method is developed to determine the optimal configuration of machines for security at an airport. The method developed was tested using 40 different devices and a configuration of 2 to 10 levels. Two Methaheuristics were used to obtain a strategy for a baggage-screening problem, with an objective function of minimizing the cost. The total cost is the sum of the Purchase cost and also the Operating cost, along with the cost of a False Alarm and False Clear. The parameter used for both (the genetic and memetic algorithm) had initial population of 100 chromosomes, and both were run 1000

times. The Genetic algorithm used the crossover and the mutation operator, while the memetic algorithm used the same parameters and in addition a local search, in this case a Tabu search. The problem was solving by creating a program for each Methaheuristic using Matlab. The best solution found is at level 9 using the following distribution of devices: 6 6 6 6 6 2 2 23 32, at a cost of 1.3618e+007 dollars. The solution says that in order to reduce costs at minimum from level 1 to level 5 use device #6, in level 6 and 7 use device #2, for level 8 use machine #23 and finally for level 10 use machine #32. This solution implies that for level one use 50 devices, at level two 26 devices, at level three 13 devices, at level four 7 devices, and finally at level six 4 devices, which means that a total of 100 of machines #6 need to be buy. Both Methaheuristics reach the optimal solution, but the memetic algorithm find the optimal solution in less iterations than the Genetic Algorithm, and also if we wan to use 9 levels or more the solutions form the genetic and memetic algorithm begin to be different, so that it is inferred that the memetic algorithm find better solutions than the genetic algorithm but with more computational effort.

References

Adbel-Magid, Y. L., Bettayeb, M., & Dawoud, M. M. (1997). "Simultaneous stabilisation of power systems using genetic algorithms". *IEEE Proceedings of Generation, Transmission and Distribution*, *144*, pp. 39-44.

Abdel-Magid, Y. L., & Abido, M. A. (August 2003). "Optimal multiobjective design of robust power system stabilizers using genetic algorithms". *IEEE Transactions on Power Systems*, *18*, pp. 1125-1132..

Alchian, A. A. (1952). "*Economic Replacement Policy*". Santa Monica: The Rand Corporation

Arkin, B. L., & Leemis, L. M. (2000). "Nonparametric Estimation of the Cumulative Intensity Function for a Nonhomogeneous Poisson Process from Overlapping Realizations". *Management Science, 46*(7), 989-998.

Barnett, A. (2004) CAPPS II: The foundation of aviation security?. *Risk Analysis*, 24(4), 909–916.

Baran, M. E., & Wu, F. F. (1989, April). "Network Reconfiguration in Distribution Systems for Loss Reduction and Load Balancing". *IEEE Transactions on Power Delivery, 4*(2).

Bellman, R. (1955). "Equipment Replacement Policy". *Journal of the Society for Industrial and Applied Mathematics,, 3*(3), 133-136.

Besten, M. d., Stützle, T., & Dorigo, M. (2000). "Ant Colony Optimization for the Total Weighted Tardiness Problem". *Proceedings of the 6th International Conference on Parallel Problem Solving from Nature*. Springer-Verlag.

Billinton, R., & Allan, R. N. (1998). "*Reliability Assesment of Large Electric Power Systems*". Boston: Kluwer Academic Publishers.

Billinton, R., & Wang, P. (1998, November). "Distrinution system reliability cost/woth analysis using analytical and sequential simulation techniques". *IEEE Transactions on Power Systems, 13*(4)..

Black, P. E. (2004, Febraury 27). "*heuristic*" *in Dictionary of Algorithms and Data Structures [online]*. Retrieved from U.S. National Institute of Standards and Technology: http://xw2k.nist.gov/dads/HTML/heuristic.html

Black, P. E. (2009, March 30). "*metaheuristic*" *in Dictionary of Algorithms and Data Structures [online]*. Retrieved from U.S. National Institute of Standards and Technology: http://xw2k.nist.gov/dads/HTML/metaheuristic.html.

Bris, R., Chatelet, E., & Yalaoui, F. (2003). "New method to minimize the preventive maintenance cost of series-parallel systems". *Reliability Engineering and System Safety, 82*, 247-255.

Brown, M. (December 1991). "*A Mean-Variance Serial Replacement Decision Model: The Independent Case*". Technical.

Brown, R. E., & Willis, H. L. (2006). "The Economics of Aging Infrastructure". *IEEE Power & Energy Magazine, 4*, 36-43

Butler, V and Poole, R.W. (2002). Rethinking Checked-Baggage Screening. Photodisc, Inc, Pg. 5-8.

Candalino, T.J; Kobza, J.E; Jacobzon, S.H. (2004). Designing optimal aviation baggage screening strategies using simulated annealing. Computers & Operations Research 31. Pg. 1753–176.

Chakrabarti S and Strauss A. (2002). Carnival Booth: An Algorithm for Defeating the Computer-Assisted Passenger Screening System. *Law and Ethics on the Electronic Frontier*.

Chang, C. F. (2008). "Reconfiguration and Capacitor Placement for Loss Reduction of Distribution Systems by Ant Colony Search Algorithm". *IEEE TRANSACTIONS ON POWER SYSTEMS, 23*(4).

Childress, S., & Cohen, P. D. (2005). "On parallel machine replacement problems with general replacement cost functions and stochastic deterioration". *Naval Research Logistics, 52*.

Chiou, J. P., Chang, C. F., & Su, C. T. (2004). "Ant Direction Hybrid Differential Evolution for Solving Large Capacitor Placement Problems". *IEEE TRANSACTIONS ON POWER SYSTEMS, 19*(4).

Chowdhury, A. A., & Koval, D. O. (1998, January/February). "Value-Based Distribution System Reliability Planning". *IEEE Transactions on Industry Applications, 34*(1).

Coit, D. W., & Smith, A. E. (June 1996). "Reliability Optimization of Series-Parallel Systems using a Genetic Algorithm". *IEEE Transactions on Reliability, 45*(2).

Coit, D.W. and Jin T. (2001). Prioritizing system-reliability prediction improvements. *IEEE Transactions on Reliability,* 50(1):17-25.

Coughlin, C. C; Cohen, J. P; and Khan, S. R. (2002). Aviation Security and Terrorism: A Review of the Economic Issues. *The Federal Reserve Bank of St. Louis*, 9-24.

Crispim, J. and Brandao J. (2005). Metaheuristics Applied to Mixed and Simultaneous Extensions of Vehicle Routing Problems with Backhauls. Journal of the Operational Research Society 56, Pg 1296-1302.

Rommert Dekker (1996), Applications of maintenance optimization models: a review and analysis. Reliability Engineering and System Safety 51 (1996) 229-240.

Dorigo, M., Maniezzo, V., & Colorni, A. (1996, February). "Ant System: Optimization by a Colony of Cooperating Agents". *Systems, Man, and Cybernetics, Part B: Cybernatics, IEEE Transactions on, 26*(1), 29-41.

E. Regnier, G. Sharp and C. Tovey (2010), Replacement under ongoing technological progress. IIE Transactions

Esogbue, A. O., & Hearnes, W. E. (1998, November). "On Replacement Models Via a Fuzzy Set Theoretic Framework". *IEEE Transactions on Systems, man, and Cybernetics - Part C: Applications and Reviews, Vol. 28, No.4*.

Espiritu, J. F., & Coit, D. W. (2008). "Component Replacement Analysis for Radial Distributions Systems". *Industrial Engineering Research Conference*.

Frederickson, H.G. and LaPorte, T.R. (2007). Airport Security, High Reliability, and the Problem of Rationality. American Society for Public Administration. Vol 62, Issue Supplement s1, Pages 33-43.

Gallego, R. A., Monticelli, J. A., & Romero, R. (2001). "Optimal Capacitor Placement in Radial Distribution Networks". *IEEE TRANSACTIONS ON POWER SYSTEMS, 16*(4).

Garg P. (2009). A Comparison between Memetic algorithm and Genetic algorithm for the cryptanalysis of Simplified Data Encryption Standard algorithm. *International Journal of Network Security & Its Applications (IJNSA)*, Vol.1, No. 1, 34-42.

Glover, F., & Laguna, M. (1997). "*Tabu Search*". Norwell, MA, USA: Kluwer Academic Publisher.

Goldberg, D. E. (1989). "*Genetic algorithms in search, optimization, and machine learning*". Addison-Wesley Pub. Co., 1989.

Hartman, J. C. (2000). "The Parallel Replacement Problem with Demand and Capital Budgeting Constraints". *Naval Research Logistics*, 40-56.

Hartman, J. C. (2004). "Multiple Asset Replacement Analysis under Variable Utilization and Stochastic Demand". *European Journal of Operations Research, 159*(1), 145-165.

Hopp, W. J., & Nair, S. K. (1991, April). "Timing equipment replacement decisions due to discontinous technological change". *Naval Research Logistics, 38*(2), 203-220.

Jacobson, S. H. and Karnani T. (2005). Assessing the impact of deterrence on aviation checked baggage screening strategies. Int. J. Risk Assessment and Management, Vol. 5, No. 1, Pg 1-14.

Kabisama, H. W. (1993). "*Electric Power Engineering*". New York: McGraw-Hill..

Kamepalli, H. B. (2001). The optimal basics for GAs. IEEE Potentials, Vol. 20, Issue 2, 25-27.

Karabakal, N., Lohmann, J. R., & Bean, J. C. (1994, March). "Parallel Replacement under Capital Rationing Constraints". *Managment Sciences, 40*(3), 305-319.

Kennedy, J., & Eberhart, R. (1995). "Particle Swarm Optimization". *IEEE International Conference on Neural Networks*, *IV*, pp. 1942-1948.

Khodr, H. M., Olsina, F. G., De Oliveira-De Jesus, P. M., & Yusta, J. M. (2008). "Maximum savings approach for location and sizing of capacitors in distribution systems". *Electric Power Systems Research, 78*.

Krasnogor, N. and Smith, J. (2005). A Tutorial for Competent Memetic Algorithms: Model, Taxonomy and Design Issues. *IEEE Transactions on Evolutionary Computation*. Vol. 9, Number 5, 474-489.

Kwang, E.Y. and Youn C.C. (2005). Analytic hierarchy process approach for identifying relative importance of factors to improve passenger security checks at airports. Vol 12, Issue 3, Pg 135-142.

Lee, A.J. and Jacobson, S.H. (2011). Evaluating the effectiveness of sequential aviation security screening policies. IIE Transactions. Vol. 43, Issue 8, Pg 547-565.

Létourneau, S., Famili, F., & Matwin, S. (1999, December). "Data mining to predict aircraft component replacement". *IEEE Intelligent Systems Journal*.

Leung, L. C., & Tanchoco, J. (1990). "Multiple Machine Replacement Analysis". *Engineering Costs and Production Economics, Vol. 20*, 265-275.

Levitin, G., Kalyuzhny, A., Shenkman, A., & Chertkov, M. (2000). "Optimal Capacitor Allocation in Distribution Systems Using a Genetic Algorithm and a Fast Energy Loss Computation Technique". *IEEE Transaction on Power Delivery, 15*(2).

Liu, C.-C., Jung, J., Heydt, G. T., Vittal, V., & Phadke, A. G. (2000, August). "The strategic power infrastructure defense system A Conceptual Design". *IEEE Control Systems Magazine*.

Masoum, M. A., Ladjevardi, M., Jafarian, A., & Fuchs, E. F. (2004). "Optimal Placement, Replacement and Sizing of Capacitor Banks in Distorted Distribution Networks by Genetic Algorithms". *IEEE TRANSACTIONS ON POWER DELIVERY, 19*(4).

Martonosi, S. E. and Barnett, A. (2006). How Effective Is Security Screening of Airline Passengers?. *Interfaces*, November/December. 36:545-552

McLay L. A; Jacobson S. H; Kobza J. E. (2004). A Multilevel Passenger Screening Problem for Aviation Security. Wiley InterScience Pg 1-15.

Mendoza, J. E; Castanier, B; Guéreta, C; Medaglia, A. L and Velasco N. (2009). A Memetic algorithm for the multi-compartment vehicle routing problem with stochastic demands. *Computers & Operations Research*. 37, 1886—1898.

Merkle, D., Middendorf, M., & Schmeck, H. (2002, August). "Ant Colony Optimization for Resource-Constraint Project Scheduling". *IEEE Transactions on Evolutionary Computation, 6*(4).

Moscato, P. (1989). On evolution, search, optimization, Genetic algorithms and martial arts: Towards Memetic algorithms. *Caltech Concurrent Computation Program*.

Nair, S. K., & Hopp, W. J. (1992). "A model for equipment replacement due to technological obsolescence". *European Journal of Operational Research, 63*, 207-221.

Natali Hritonenko, Yuri Yatsenko (2006), Optimal equipment replacement without paradoxes: A continuous analysis. Operations Research Letters 35 (2007) 245–250

National Research Council, Airline Passenger Security Screening, National Academy Press Publication NMAB- 482-1, Washington, DC, 1996.

O'Harrow R. Jr. (2002) Intricate Screening Of Fliers In Works Database Raises Privacy Concerns. Washington Post. A01.

Pansini, A. J., & Smalling, K. D. (2002). "*Guide to eletric power generation*". Georgia, USA: The Fairmont Press, Inc.

Ping-Teng Chang (2003). Fuzzy strategic replacement analysis. European Journal of Operational Research 160 (2005) 532–559.

Poole, Jr., R.W. and Passantino, G. (2003) A risk-based airport security policy. Technical report, *Reason Public Policy Institute*, Public Policy no. 308, Los Angeles, CA.

Rao, E., Dickey, R., 1999, "Evaluation of the FAA's Security Equipment Integration Program," International Carnahan Conference on Security Technology, 158-167.

Rajagopalan, S. (1998, November). "CAPACITY EXPANSION AND EQUIPMENT REPLACEMENT: A UNIFIED APPROACH". *Operations Research, 46*(6).

Samrout, M., Yalaoui, F., Chatelet, E., & Chebbo, N. (2005). "New methods to minimize the preventive maintenance cost of series-parallel systems using ant colony optimization". *Reliability Engineering and System Safety, 89*, 346-354.

Sand, K., Seljeseth, H. and Samdal, K. (2004). Quality of supply regulation - status and trends. Nordic distribution and asset management conference.

Singh Sameer and Singh Maneesha (2003). Explosives Detection Systems (EDS) for Aviation Security: A Review. PANN Research, Pg 8-26.

Sethi, S., & Chand, S. (1979). "Planning Horizon Procedures for Machine Replacement Models". *Management Science, Vol. 25, No. 2*, 140-151.

SGI Federal (2003). Enabling Effective Power Management Solutions for Today's Changing Power Environment. http://www.sgi.com/pdfs/2999.pdf

Stützle, T. G. (1999). "*Local search algorithms for combinatorial problems: analysis, improvements, and new applications*". Infix.

Sundhararajan, S., & Pahwa, A. (1994). "Optimal selection of capacitors for radial distribution systems using genetic algorithm". *IEEE Transactions on Power Systems, 9*(3).

Taleski, R., & Rajicic, D. (1997, February). "Distribution network reconfiguration for energy loss reduction". *IEEE Transactions on Power Systems, 12*(1).

Virta J. L; Jacobson S. H; Kobza J. E. (2003). Analyzing the Cost of Screening Selectee and Non-Selectee Baggage. Risk Analysis, Vol. 23, No. 5. Pg 897-908.

Weiss, W.E. (2011). Development and use of the dynamic security model for airports. *Journal of Airport Management*. Vol. 5, Issue 3, 245-254.

Xiaofeng, N; Batta, R; Drury C.G; Li L; (2009). The impact of Joint Responses of
Devices in an Airport Security System. Risk Analysis: An International Journal. Vol 29, Issue 2. Pg 298-311.

Yamayee , Z., Sidenblad, K., & Yoshimura, M. (1983, February). 'A COMPUTATIONALLY EFFICIENT OPTIMAL MAINTENANCE SCHEDULING METHOD". *IEEE Transactions on Power Apparatus and Systems, 102*(2).

Yangpin, Z., Bingquan, Z., & DongXin, W. (2000). "Application of genetic algorithms to fault diagnosis in nuclear power plants". *Reliability Engineering and System Safety, 67*, 153-160.

Yano, C. A. (1984). "*On the equivalence of an equipment replacement problem and a facility location problem*". Technical Report 84-27, The University of Michigan, Department of Industrial & Operations Engineering, Ann Arbor.

Ziari, I., Ledwich, G., Ghosh, A., Cornforth, D., & Wishart, M. (2010). "Optimal allocation and sizing of capacitors to minimize the transmission line loss and to improve the voltage profile". *Computers and Mathematics with Applications, 60*, 1003-1013.

Zionts, S. (1974). "*Linear and integer programming*". Englewoods Cliffs, NJ, USA: Prentince-Hall, Inc.

Vita

Anuar was born in Guadalajara, Jalisco, Mexico. He received (B.E.) Bachelor of Engineering degree in Industrial Engineering from the University of Texas at El Paso, United States in December 2010. Since January 2011, he has been working towards his (M.S) Master of Science degree in industrial Engineering specializing in reliability and optimization techniques. During his master's degree program at UTEP, he got an opportunity to work with his graduate advisor Dr. Jose F. Espiritu, as Graduate Research Assistant and Graduate Teaching Assistant. Since then he has been working on his research topic concurrently assisting Dr. Espiritu in his research areas of power systems optimization using various evolutionary algorithms.

Permanent address: Calzada del Parque #518, Col. Margaritas
Ciudad Juarez, Chihuahua, Mexico

This thesis/dissertation was typed by Anuar Jesús Aguirre Nuñez