Studies in Heuristics for the Annual Crop Planning Problem

by

Sivashan Chetty

Submitted in fulfilment of the academic requirements
for the degree of Master of Science in Computer Science

at the

School of Mathematics, Statistics and Computer Science
University of KwaZulu-Natal
Durban, South Africa

December 2012

Examiner's Copy

UNIVERSITY OF KWAZULU-NATAL

COLLEGE OF AGRICULTURE, ENGINEERING AND SCIENCE

DECLARATION

The research described in this thesis was performed at the University of KwaZulu-Natal under the supervision of Dr. A. O. Adewumi. I hereby declare that all materials incorporated in this thesis are my own original work except where acknowledgement is made by name or in the form of a reference. The work contained herein has not been submitted in part or whole for a degree at any other university.

Signed:

Sivashan Chetty

Date: December 2012

As the candidate's supervisor, I have approved the dissertation for submission

Signed:

Dr. A. O. Adewumi

Date: December 2012

UNIVERSITY OF KWAZULU-NATAL

COLLEGE OF AGRICULTURE, ENGINEERING AND SCIENCE

DECLARATION – PLAGIARISM

I, ____ declare that

1. The research reported in this thesis, except where otherwise indicated, is my original research.
2. This thesis has not been submitted for any degree or examination at any other University.
3. This thesis does not contain other persons' data, pictures, graphs or other information, unless specifically acknowledged as being sourced from other persons.
4. This thesis does not contain other persons' writing, unless specifically acknowledged as being sourced from other researchers. Where other written sources have been quoted, then:
    a. Their words have been re-written but the general information attributed to them has been referenced
    b. Where their exact words have been used, then their writing has been placed in italics and inside quotation marks, and referenced.
5. This thesis does not contain text, graphics or tables copied and pasted from the Internet, unless specifically acknowledged, and the source being detailed in the thesis and in the References sections.

Signed:

____

Sivashan Chetty

DEDICATION

To my family

ACKNOWLEDGEMENTS

Without God I would not have had the ability, strength or wisdom to complete this research. I thank God for the success of this research and for more successes to come.

I am grateful to my family. I thank you all for your love and encouragement. You have given me strength by supporting me in all I do. I appreciate you.

I am very grateful to my supervisor Dr. Aderemi O. Adewumi. Thank you Sir, you are exceptional. Thank you for your encouragement and assistance with this research. You have helped me believe in myself and my abilities. For this I am forever grateful.

My thanks also go to all those that believe in my abilities and that encourage, strengthen and support me continually. God bless you all.

ABSTRACT

Increase in the costs associated with agricultural production and the limited availability of resources have amplified the need for optimized solutions to the problem of crop planning. The increased costs have imparted negatively on both the cost of production as well as the sale prices of finished products to consumers, with the resultant effects on the socio-economic livelihoods of people around the world. This has increased the burden of poverty, malnutrition, diseases and other types of social problems. The limited availability of land, irrigated water and other resources in crop planning therefore demand optimal solutions to the problem of crop planning, in order to maintain the desired level of profitable outputs that do not strain available resources while still meeting the demands of consumers. Incidentally, the current situation is such that crop producers are required to generate more output per area of crops cultivated within the ambit of the available resources for crop production. This creates a great challenge both for farmers and researchers. Interesting, the problem is essentially an optimization problem hence a challenge to researchers in mathematical and computing science.

Notably within the agricultural sector, achieving efficient use of irrigated water demands that optimized solutions be found for its usage during crop planning and production. Incidentally, increase in population growth and limited availability of fresh water has increased the demand of fresh water supply from all sectors of the economy. This has increased the pressure on the agricultural sector as being one of the primary users of fresh water supply to use irrigated water more efficiently. This is to minimize excessive water wastage. It has therefore become very important that optimized solutions be found to the allocation and use of the irrigated water, for water conservational purposes. This is also a very essential key to crop planning decisions.

Therefore, in order to determine good solutions to crop planning decisions, this study dwells on a fairly new but important area of agricultural planning, namely the Annual Crop Planning (ACP) problem which essentially focuses at the level of an irrigation scheme. The study presents a model of the ACP problem that helps to determine solutions to resource allocations

amongst the various competing crops that are required to be grown at an irrigation scheme within a year. Both new and existing irrigation schemes are considered.

Determining solutions for an ACP problem requires that the requirements and constraints presented by crop characteristics, climatic conditions, market demand conditions and the variable costs associated with agricultural production are observed. The objective is to maximize the total gross profits that can be earned in producing the various crops within a production year.

Due to the complexity involved in determining solutions for an ACP problem, exact methods are not researched in this study. Rather, to determine near-optimal solutions for this NP-Hard optimization problem, this research introduces three new Local Search (LS) metaheuristic algorithms. These algorithms are called the Best Performance Algorithm (BPA), the Iterative Best Performance Algorithm (IBPA) and the Largest Absolute Difference Algorithm (LADA). The motivation for implementing these algorithms is to investigate techniques that can be used to determine effective solutions to difficult optimization problems at low computational costs.

This study also investigates the performances of three recently introduced swarm intelligence (SI) metaheuristic algorithms in determining solutions to the ACP problems studies. These algorithms have shown great strength in providing competitive solutions to similar optimization problems in literature, hence their use in this work. To the best of the researchers' knowledge, this is the first work that reports comparative study of the performances of these particular SI algorithms in determining solutions to a crop planning problem. Interesting results obtained and reported herein show the viability, effectiveness and efficiency of incorporation proven metaheuristic techniques into any decision support system that will help determine solutions to the ACP problem.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ABBREVIATIONS

Nomenclature	Definitions
ACP	Annual Crop Planning
LS	Local Search
BPA	Best Performance Algorithm
IBPA	Iterative Best Performance Algorithm
LADA	Largest Absolute Difference Algorithm
SA	Simulated Annealing
TS	Tabu Search
CS	Cuckoo Search
FA	Firefly Algorithm
GSO	Glowworm Swarm Optimization
GA	Genetic Algorithm
SI	Swarm Intelligence
SAP	Space Allocation Problem
EA	Evolutionary Algorithm
DE	Differential Evolution
ACO	Ant Colony Optimization
CWR	Crop Water Requirement
VIS	Vaalharts Irrigation Scheme
TIS	Taung Irrigation Scheme
PL	Performance List
SL	Solutions List
TL	Tabu List

OUTCOME OF RESEARCH WORK (PUBLICATIONS)

Article Already Published in Peer-reviewed ISI Journals

1. S. Chetty and A.O. Adewumi (2013), "Comparison Study of Swarm Intelligence Techniques for the Annual Crop Planning Problem," *IEEE Transactions on Evolutionary Computation,* DOI 10.1109/TEVC.2013.2256427, Online first, issue 99.
2. S. Chetty and A.O. Adewumi (2013), "Three New Stochastic Local Search Metaheuristics for the Annual Crop Planning Problem Based on a New Irrigation Scheme," *Journal of Applied Mathematics*, Vol. 2013, Article ID 158538, 14 pages
3. S. Chetty and A.O. Adewumi (2013), "Three New Stochastic Local Search Algorithms for Continuous Optimization Problems," *Computational Optimization and Applications*, Springer US, Online First, DOI 10.1007/s10589-013-9566-3, 47 pages, May.

Article published in Peer-reviewed Conference Proceedings

4. S. Chetty and A.O. Adewumi (2012), " Results of Local Search Heuristics for the Annual Crop Planning Problem," Proceedings of the 41st Annual Conference of the Operations Research Society of South Africa (ORSSA 2012), September 16 - 19, Pretoria, South Africa. pp. 83-92.

Article submitted to peer-reviewed ISI Journals

5. S. Chetty and A.O. Adewumi, "Studies in Swarm Intelligence Techniques for Annual Crop Planning Problem in a New Irrigation Scheme". Submitted to the South African Journal of Industrial Engineering (revised version submitted).
6. S. Chetty and A.O. Adewumi, "On the Performance of New Local Search Heuristics for Annual Crop Planning: Case study of the Vaalharts Irrigation Scheme." Submitted to the Journal of Experimental & Theoretical Artificial Intelligence.

CHAPTER ONE
INTRODUCTION AND SCOPE

1.1 Background and Motivation

Recently, increased costs associated with agricultural production coupled with a limited availability of production resources have amplified the need for optimized solutions to the problem of crop planning. Expectedly, the increased costs associated with crop production have resulted in increases in the price of food products which have had negative effects on the standards of living of people especially in sub-Saharan Africa. Thus, the increased prices of food coupled with the shortages of food supply have contributed to various forms of social and economic problems including poverty, disease and malnutrition. This puts more pressure on farmers especially crop producers. At present, crop producers are required to make more efficient decisions in managing their limited resources for crop production. In spite of the limited agricultural resources, it is also becoming increasingly very important for crop producers to simultaneously raise the returns achieved per area of crops cultivated.

According to the Food and Agriculture Organization of the United Nations, it is now estimated that more than a billion people suffer from under-nourishment (FAO, 2010) which is a reflection of the state of things in the agricultural sector as the primary supplier of food (Schmitz *et al*., 2007). Therefore efforts to combat the problems of increased production costs, increased food prices, shortages in food supply, poverty and starvation must also focus on developing optimal production of food crops within the agricultural sector.

Determining optimized solutions in crop planning is a complex and difficult problem. Aside the fact that crop production involves multi-stage processes, there are several competitive and conflicting factors that must be taken into consideration. Some of these factors are predictable while others are stochastic. However, all factors are important and will eventually have impact on the different stages of the crop production process. The multiple stages of the crop production process include crop selection, land allocations, planting, the growth stages, harvesting, crop storage and the marketing stage (Acquaah, 2004). Each stage of the crop production process will therefore require careful planning. Planning is important as the decisions made at each stage will influence the outcome determined at the end of the cropping season for each crop and at the end of a production year for all crops.

During the crop selection process, the factors associated with the geographical location of the farm, crop characteristics, production costs and the uncertainty of operating within a deregulated marketing environment have major influence on the decisions made in selecting the crops to be cultivated. At a specific geographical location, the climatic and soil factors are important. This will determine the types of crops that will most suitably adapt to the given geographical location (Mustafa *et al*., 2011). In terms of crop characteristics, the crops' water requirements and the crops' yield are important. The market demand and supply conditions and the production costs will also influence the selling prices of the harvests.

Once the crop selection has been finalized, solutions will need to be determined in allocating a limited area of agricultural land amongst the various competing crops to be planted or cultivated. In allocating land, the crop yields, forecasted market prices, market demand conditions and the various costs associated with crop production need to be considered. The main objective for determining optimal land allocation is to maximize the total gross profits that can be earned in the production and sale of the harvests. Similarly, during the planting process, crop growth and the harvesting stages the limited resources available for crop production will need to be efficiently allocated. The limited resources include labor, equipment, fertilizers, pesticides and irrigated water, among others. During the crop growth stage, the limited resources will need to be allocated to the different crops according to their daily needs. It is important that close attention be paid during the crop growth stage of the crop production process (Dukes *et al*., 2012). Meanwhile, several factors can hinder plant growth which will then ultimately affect the yield. Another challenge that comes up after crops have been harvested is to determine solutions that minimize storage costs and also to determine the best marketing strategy that maximizes the total gross profits earned in the sale of harvests.

There are several uncertain factors that need to be considered in making crop planning decisions (Astera, 2012). These uncertain factors that influence crop production are the climatic conditions, the soil characteristics, the forecasted market prices and the cultivation practices, amongst others. The climatic conditions include factors such as rainfall, temperature and drought. The soil conditions include the nutritional quality of the soil, the soil texture, the soil moisture balance and its drainage systems, among others (Astera, 2012). Since the exact selling prices of the crops are not known in advance, forecasted selling prices are used to determine the area of land under which the crops should be cultivated (Kantanantha, 2007). Furthermore, cultivation practices have a major influence on the crop's growth stages, and the yields produced. In cultivation, weeds, pests and bacteria must be catered for. If crop producers knew these uncertain factors in advance, it would allow for better preparation for the production year ahead (Kantanantha, 2007).

Determining optimized solutions for the different stages of the crop production process has attracted considerable research in different academic disciplines. Due to the complexity of these problems and the uncertainty of several factors, there are no methods that exist that guarantee optimal solutions in crop planning. The main aim is therefore to find the best possible solutions within reasonable computational time, given the probable rainfall patterns, the costs of crop production, the crop yields, the market demand and supply conditions and the forecasted producer prices. The solutions found then serves to advise crop planners on the best way to go about resource allocations amongst the various competing crops that are required to be produced within a production year.

This research focuses on determining solutions to the land allocation problem of the crop production process, specifically at the level of an irrigation scheme. At this level, suggestions can be made concerning resource allocations amongst various competing crops that are required to be grown within a production year. The objective of determining the best resource allocations will be to optimize the total gross profits that can be earned from all crops produced within a year. Therefore, to help in determining optimal solutions in allocating land, irrigated water supply and the variable costs associated with crop production, a new model for the Annual Crop Planning (ACP) problem has been introduced as part of the agricultural planning problem in this study. To the best of the researchers' knowledge, this is the first attempt to define and model ACP as an optimization problem as presented in this research study. Two mathematical models for the ACP problems for both new and existing irrigation schemes were developed. The study uses the Vaalharts and Taung Irrigation Schemes, which are neighboring irrigation schemes located at the borders of the Northern Cape and North West Province of South Africa (Grove, 2008), as case studies.

Like many similar real-world optimization problems, the ACP problem is *NP*-Hard in nature. Generally, various types of optimization problems in literature have been solved using exact or heuristic (approximate) methods (Adewumi and Ali, 2010). Exact methods guarantee that the optimal solution will be found. However, for *NP*-Hard problems, exact methods do not guarantee that the optimal solution will be found within reasonable computational time (Trevisan, 2011). Exact methods are preferred for optimization problems where the optimal solution can be determined within polynomial time (P). However, if the computational time involved with determining the optimal solution increases exponentially then exact methods are not preferred. The complexities of many real-world optimization problems, like the ACP problem, have therefore made the use of exact methods in providing solutions a rare occurrence. Rather, researchers have settled for near-optimal solutions that compromise accuracy for speed with the use of heuristic approaches. Efforts are currently geared towards providing 'intelligent' heuristic solutions to complex optimization problems. The majority of these intelligent algorithms are developed and modeled after some natural processes or behaviors of animals in nature. Examples include the modeling of the annealing process that occurs when heated metal begins to cool, modeling the social behavior of swarms of biological agents and using memory ability, amongst others. Heuristic algorithms that use more advanced techniques in determining solutions are referred to as metaheuristic algorithms.

To determine solutions to the ACP problems for new and existing irrigation schemes, this research has investigated the usefulness of employing both Local Search (LS) and Swarm Intelligence (SI) metaheuristic algorithms. LS metaheuristic algorithms make slight changes to the solutions being worked with in trying to determine improved solutions in an iterative way. SI algorithms are population-based algorithms that model the way biological agents interact with each other and their environments in accomplishing an overall task (Blum and Merkle, 2008). This research introduces three *new* LS metaheuristic algorithms, and investigates three relatively new SI metaheuristic algorithms in an effort to determine solutions to the ACP problems studied in this research. Generally, both LS and SI metaheuristics have been successfully used to determine solutions to many real-world NP-Hard optimization problems.

In terms of land allocation, the ACP problem was considered and modeled as an instance of the Space Allocation Problem (SAP) (Adewumi, 2010; Adewumi and Ali, 2010; Silva, 2003). SAP's are amongst the hardest optimization problems found in literature (Silva, 2003). Space allocation involves allocating a limited area of available space amongst a finite number of demanding entities that require space utilization, under given constraints and requirements (Silva, 2003). The objective is to determine a solution that allocates the limited area of available space in such a way that provides the best level of satisfaction amongst all demanding entities. Instances of SAP's in literature include shelf space allocation (Tsai and Wu, 2010; Bai, 2005), office space allocation at tertiary institutions (Silva, 2003) and the hostel space allocation (Adewumi and Ali, 2010), amongst others. Some of these instances of SAP's have been modeled mathematically as variants of known benchmark discrete optimization models such as bin-packing, assignment modeling, and knapsack modeling (Silva, 2003). The ACP problems introduced in this work have been modeled using a modified form of the knapsack model. Specifically, a bounded-fractional-multiple knapsack model with an added constraint has been used.

1.2 Contributions of this Thesis

This research work makes the following contributions:

1. The description of the Annual Crop Planning (ACP) problem at the level of an irrigation scheme is presented. The ACP problems presented are those at new and existing irrigation schemes.
2. Two practical mathematical models are introduced for determining solutions to these ACP problems.
3. An investigation into the suitability of employing both Local Search (LS) and Swarm Intelligence (SI) metaheuristic algorithms, in determining solutions to these ACP's has been done. Comparisons of the performances of both the LS and SI algorithms are done. This research shows that the LS and SI metaheuristic algorithms can successfully be applied in providing competitive solutions to crop planning problems.
4. Three new LS metaheuristic algorithms have been introduced. The performances of the new LS metaheuristic algorithms are shown to be very competitive in determining solutions.
5. For the first time, a comparative study in the performances of the Firefly Algorithm, Cuckoo Search and Glowworm Swarm have been made in determining solutions to a crop planning problem.
6. In addition to the available data used, twelve new test datasets have been compiled and are made available to further encourage research to the problem of ACP.

1.3 Overview of this Thesis

The remainder of this dissertation is as follows:

Chapter two discusses the field of optimization. Classifications of the different types of optimization problems are presented. Techniques used in determining solutions to optimization problems are also discussed.

Chapter three discusses the multi-stage process of crop production. Attention is paid to the various costs associated with the production process. The several factors that affect the plant growth and its yield are discussed. A description of the conditions associated with the geographical location of the case studies in this research is given. The problem definition is also formalized.

Chapter four presents the formulation of the ACP problem as a Space Allocation Problem. The formulation of the ACP mathematical model is also described.

Chapter five presents and describes the three new LS metaheuristic algorithms. Descriptions of two other popular LS metaheuristic algorithms are also given. These algorithms will be used to compare the performances of the new LS algorithms in their abilities to determine solutions.

Chapter six presents and describes three recently developed SI metaheuristic algorithms. The description of a well-known population based metaheuristic algorithm is also given. This algorithm is used to compare the performances of the SI algorithms in their abilities to determine solutions.

Chapter seven presents the ACP mathematical model used for determining solutions to the ACP problem at an existing irrigation scheme. The solutions determined by the LS and population based metaheuristic algorithms are also presented and discussed. Conclusions are drawn concerning the possible strengths and weaknesses in determining solutions to the ACP problem at an existing irrigation scheme.

Chapter eight presents the ACP mathematical model used for determining solutions to the ACP problem at a new irrigation scheme. Similar to chapter seven, the solutions determined by the LS and population based algorithms are presented and discussed. Conclusions are also drawn concerning the possible strengths and weaknesses of the algorithms in determining solutions to the ACP problem at a new irrigation scheme.

Finally, chapter nine draws conclusions and discusses possible future research work.

CHAPTER TWO
AN OVERVIEW OF GLOBAL OPTIMIZATION

2.1. Introduction

Optimization problems exist all around us. There is always a desire to determine the optimal solution in accomplishing a task. Simple examples of optimization problems range from finding the shortest walking distance between any two points, to minimizing the distance travelled by hundreds of vehicles in trying to optimize fuel consumption, amongst others. Optimization is therefore a very relevant field of study which has attracted enormous interest academically. It is largely studied in the fields of Computer Science, Mathematics and Economics, amongst others (Boyd and Vandenberghe, 2004). The goal in determining solutions to optimization problems is to determine a solution that will optimize the problems' objective. The solution found must exist within the domain of the solution space. For the solution found to be feasible, it must satisfy the multiple constraints and objectives that are associated with the objective function.

This chapter briefly describes the field of optimization. Attention is paid to the different categories of optimization problems and the techniques used to determine solutions.

2.2. Mathematical Optimization

A formal definition of optimization is as follows (Snyman, 2005):

Definition 2.1: Let $f: A \rightarrow \mathbb{R}$ represents an objective function. $A \subset \mathbb{R}$ is a set of feasible solutions that exist within the solution space of real numbers $\mathbb{R}$. Let $x^* \in A$. The objective is to determine $x^* \in A \subset R$ such that $f(x^*)$ either minimizes or maximizes the objective function f, i.e.

$$f(x^*) \leq f(x), \forall\, x \in A \text{ (minima)} \tag{2.1}$$

$$f(x^*) \geq f(x), \forall\, x \in A \text{ (maxima)} \tag{2.2}$$

In equations 2.1 and 2.2, $f(x^*)$ are optimal solutions. Optimal solutions are found within the local neighborhood structures of a solution space.

A neighborhood structure is defined as follows (Blum and Roli, 2003):

Definition 2.2: Let $\mathbb{N}: S \rightarrow 2^S$ be a function that assigns to every feasible solution $i \in S$ a subset of feasible solutions $j \in \mathbb{N}(i) \subseteq S$. $\mathbb{N}(i)$ is called the neighborhood of solution i if each neighbor $j \in \mathbb{N}(i)$ is in some way close to i within the domains of the solution space S.

Optimal solutions that are found within the local neighborhood structures of a solution space are called the local optima. The local optima can either be the local minimum or maximum solutions. Several local optimum solutions may exist within the local neighborhood structures of a solution space. The best local minima or maximum that exists within the solution space is the global optimal solution. Global optimal solutions are local optimal solutions, but not necessarily vice versa.

The definition of local minimum, local maximum, global minimum and global maximum solutions are given by definitions 2.3, 2.4, 2.5 and 2.6 below. For these definitions let f represent an objective function and let $\mathbb{R}$ represent a solution space of real numbers (Hancock, 2005).

Definition 2.3: A local minimum exists at a point $x^* \in \mathbb{R}$ if there exists some value $\varepsilon > 0$ such that

$$f(x^*) \leq f(x), \text{ subject to } |x - x^*| < \varepsilon, \forall\, x \in \mathbb{R} \tag{2.3}$$

In equation 2.3, $|x - x^*|$ is the absolute value of the difference between x and x^*.

Definition 2.4: A local maximum exists at a point $x^* \in \mathbb{R}$ if there exists some value $\varepsilon > 0$ such that

$$f(x^*) \geq f(x), \text{ subject to } |x - x^*| < \varepsilon, \forall\, x \in \mathbb{R} \qquad (2.4)$$

Definition1.5: A global minimum exists at a point $x^* \in \mathbb{R}\ iff$,

$$f(x^*) \leq f(x), \forall\, x \in \mathbb{R} \qquad (2.5)$$

Definition 1.6: A global maximum exists at a point $x^* \in \mathbb{R}\ iff$,

$$f(x^*) \geq f(x), \forall\, x \in \mathbb{R} \qquad (2.6)$$

Figure 2.2.1 illustrates local optimum solutions. The global optimal solutions are the extreme local optimum solutions. Local optimum solutions are the optimal solutions found within the local neighborhood structures of the solution space.

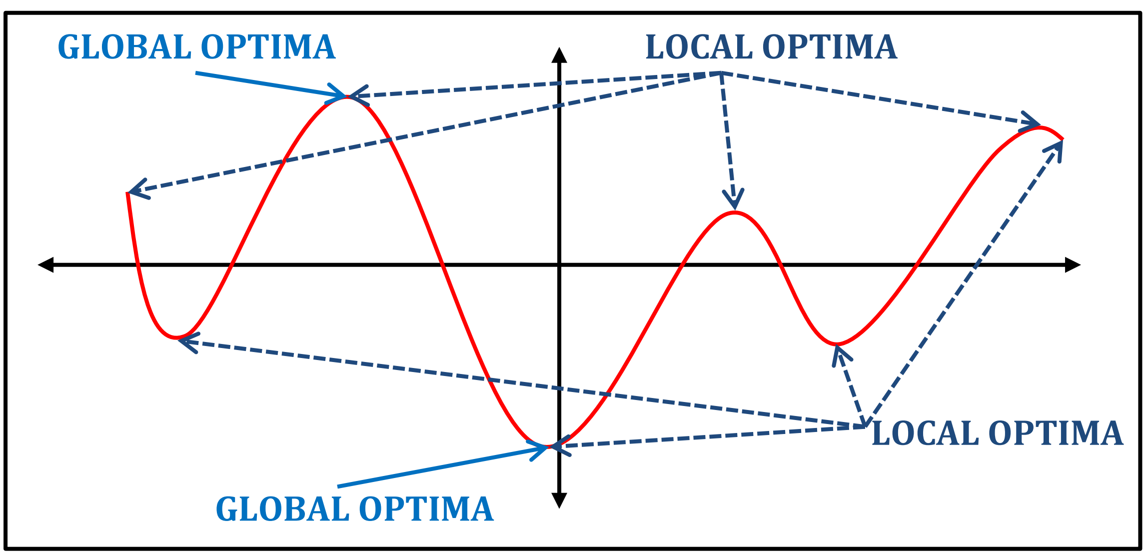

Figure 2.2.1: Local optimum solutions of a one-dimensional objective function

Determining the global optimal solution requires performing an exhaustive search of the solution space. If the global optimal solution can be found within polynomial time (P) then the solution is considered deterministic and is traceable. Deterministic solutions have a clear relationship between the optimal solution and the decision variables used to determine the optimal solution.

The computational time involved with determining the global optimal solution is not a major factor if the solution can be determined within P. However, if the computational time increases exponentially in determining the global optimal solution then computational time does become important. If only exponential time algorithms exist in determining the global optimal solution then the problem is considered intractable and is non-deterministic polynomial (NP) (Silva, 2003). For NP type optimization problems, performing an exhaustive search of the solution space may be infeasible. For these types of problems, accepting approximate solutions is more widely acknowledged. Approximate or near-optimal solutions are not the global optimal solutions but are considered acceptable if the solutions can be found within P, for NP type optimization problems.

There are two types of methods used to determine solutions to optimization problems. These include exact and heuristic algorithms. Exact algorithms exhaustively search the solution space in order to determine the global optimal solution. These algorithms don't consider the computational time involved with determining the global optimal solution (Trevisan, 2011). Since many real-world optimization problems are NP in nature, exact algorithms are not preferred in determining solutions if the computational time is expected to be exponential. Examples of exact algorithms include Linear Programming, Dynamic Programming and Branch and Bound.

Heuristic algorithms provide near-optimal solutions to optimization problems. Near-optimal solutions are accepted when no polynomial-bound algorithm exists to determine the global optima. These solutions are slightly inferior solutions but are accepted in trading accuracy for a reduction in computational time complexity (Syam and Al-Harkan, 2010). Heuristic algorithms are decision algorithms which use trial and error techniques in performing a search of the solution space. It is successfully applied in providing solutions to both continuous and combinatorial optimization problems.

2.3. Classifications of Optimization Problems

Optimization problems are classified in many ways. The classifications are based on the problem constraints, the nature of the equations involved, the number of objective functions, the deterministic nature of the problem and the type of decision variables used, amongst others (Raju and Kumar, 2010). No single optimization algorithm exists that can provide optimized solutions to all types of optimization problems. Certain types of optimization techniques will therefore be more adaptable to some types of optimization problems rather than others. Brief descriptions of the primary classifications of optimization problems are given below.

2.3.1. Classification Based on Constraints

Constraints are the restrictions associated with the objective function f. They define the bounds within which feasible solutions are found. Constraints can be classified as being either hard or soft (Domshlak *et al.*, 2006). Hard constraints are those constraints that must not be broken. Soft constraints are those constraints that can be compromised. Feasible solutions that are found within the solution space are those that satisfy all hard constraints and satisfy as many soft constraints as possible.

The categorization of optimization problems, based on constraints, depend on the number of constraints that are associated with the problems' objective f. There are two types of categories. These include unconstrained and constraint optimization problems.

Unconstrained Optimization Problems: If no constraint governs the evaluation of f then the problem is an unconstrained optimization problem.

Constrained Optimization Problems: If constraints govern the evaluation of f then the problem is a constrained optimization problem.

Most real-world optimization problems are multi-constrained, however, several unconstrained optimization problems do exist.

2.3.2. Nature of the Equations Involved

The nature of the equations of the objective function f, and its constraints, can be linear, non-linear, geometric or quadratic, amongst others (Raju and Kumar, 2010). Optimization problems are therefore also classified based on the nature of the equations involved.

Linear Programming Problems (LPP): If the formulation of the optimization problem is governed by linear equations of non-negative decision variables then the problem is a LPP. Mathematically, LPP's are formulated as follows (Raju and Kumar, 2010):

Optimize: $$f(x) = \sum_{i=1}^{n} c_i x_i \tag{2.7}$$

Subject to: $$\sum_{i=1}^{n} a_{ij} x_i = b_j\,, \qquad \forall\, j = 1,2,\dots,m \tag{2.8}$$

$$x_i \geq 0 \qquad \forall\, i = 1,2,\dots,n \tag{2.9}$$

Where: c_i, a_{ij} and b_j are constants

Non-Linear Programming Problems (NLPP): If one or more constraints governing the formulation of the optimization problem are non-linear, or if f is non-linear, then the problem is a NLPP. NLPP's are the most common programming problems encountered and are mathematically represented as follows (Jain and Singh, 2003).

Optimize: $$f(x) \tag{2.10}$$

Subject to: $$h_i(x) = 0 \qquad \forall\, i = 1,2,\dots,m \tag{2.11}$$

$$g_i(x) \leq 0 \qquad \forall\, i = (m+1),\dots,p \tag{2.12}$$

Where: $h_i(x)$ = equality constraints

$g_i(x)$ = inequality constraints

$\{x_1, x_2, \ldots, x_p\}$ = design variables

Geometric Programming Problems (GMPP): If the constraints governing the formulation of the optimization problem are polynomials of the variables x then the problem is a GMPP.

Quadratic Programming Problems (QPP): These are maximization type NLPP's. They have 'concave' objective functions and linear constraints.

2.3.3. Number of Objective Functions

There may be single or multiple objective functions associated with the formulation of the optimization problem.

Single-objective Programming Problems: This type of optimization problem only requires one objective function f that would need to be evaluated.

Multi-objective Programming Problems (MPP): This type of optimization problem requires that more than one objective function be simultaneously evaluated. Most real-world problems are MPP in nature. MPP is mathematically represented as follows.

$$\text{Optimize:} \quad f_i(x) \quad \forall\, i = 1, \ldots, k \tag{2.13}$$

$$\text{Subject to:} \quad g_{ij}(x) \leq 0 \qquad \forall\, j = 1, \ldots, p_{ij} \tag{2.14}$$

2.3.4. Deterministic Nature of the Problem

The deterministic nature of an optimization problem relates to the computational time involved with determining the optimal solution. If the optimal solution can be found within P then the optimization problem is considered deterministic. If the optimal solution cannot be determined within P then the problem is considered non-deterministic. Exact methods are used to provide solutions to deterministic optimization problems. Examples of exact methods include the Divide and Conquer and the Branch and Bound algorithms. For non-deterministic type optimization problems, heuristic algorithms are preferred. Examples of metaheuristic algorithms include the Genetic Algorithm (GA), Simulated Annealing (SA) and Tabu Search (TS).

2.3.5. Type of Decision Variables Used

Decision variables can either be values taken from a real numbered system $\mathbb{R}$, or from a set of discrete values. Discrete values are the unique inputs that are allowed to be used as the decision variables to the objective function. Based on the decision variables used, optimization problems can be classified as being either continuous or combinatorial in nature.

Based on the categories mentioned above, in subsections 2.3.1 to 2.3.5, an example of an optimization problem can be that of a multi-constrained, multi-objective, linear and non-deterministic optimization problem, which may use continuous values as the decision variables to the objective function f. As also mentioned previously, the types of techniques used to provide solutions to these optimization problems include exact and heuristic methods. The explanation of exact methods is out of the scope of this research. However, heuristic algorithms are explained in subsection 2.4 below.

2.4. Heuristics Algorithms

Heuristic algorithms are suitably used to provide near-optimal solutions to NP type optimization problems, within P. They are decision algorithms which use trial and error techniques in deciding on the next solution to exploit within the local neighborhood structures of a solution space. Heuristic algorithms are iterative algorithms which usually stop after a specified number of iterations have completed or when a stopping criteria has been satisfied.

One problem of applying heuristic algorithms is premature convergence (Rocha and Neves, 1999). Premature convergence occurs when the heuristic algorithm converges to a local optimum solution, which is not close enough to the global optimal solution. To minimize the probability of premature convergence, heuristic algorithms employ more ‘intelligent’ techniques in determining solutions. These intelligent techniques allow for a more effective exploration and exploitation of the solution space.

Exploration involves exploring the neighborhood structures of the solution space to try and determine more promising areas. These promising areas may possibly contain the global optimum solution. Exploitation involves exploiting the local neighborhood structures of these promising areas in order to try and find the local optimum solution. Finding a good balance between exploration and exploitation means that an algorithm should quickly determine promising areas within the solution space but should not spend too much of time searching for the local optimum solution (Syam and Al-Harkan, 2010).

Intelligent techniques which allow for more effective exploration and exploitation of the solution space reduce the risk of premature convergence. Some intelligent techniques employed include the use of memory abilities, learning from other ‘agents’ and randomly jumping to other neighborhood structures within the solution space. Heuristic algorithms that use more intelligent techniques are called metaheuristic algorithms. Metaheuristic algorithms are not problem specific algorithms. Metaheuristic algorithms which use randomization in determining solutions fall under a category of algorithms known as the Monte Carlo algorithms (Krauth, 1998).

Popular Monte Carlo metaheuristic algorithms that provide near-optimal solutions to NP type optimization problems include Evolutionary Algorithms (EAs), Swarm Intelligence (SI), Simulated Annealing (SA) and Tabu Search (TS), amongst others. EAs include algorithms such as the GA and Differential Evolution (DE) (Storn and Price, 1997; Price *et al*., 2005). SI includes algorithms such as the Ant Colony Optimization (ACO) (Dorigo, 1992; Dorigo and Gambardella, 1997), Cuckoo Search (CS), Firefly Algorithm (FA) and Glowworm Swarm Optimization (GSO). GA, SA, TS, CS, FA and GSO are the algorithms investigated in this dissertation and are therefore referenced and explained in more detail in chapters five and six.

2.5. Conclusion

This chapter describes the field of optimization and shows its relevance in research. There are several types of optimization problems that exist in nature. Brief descriptions have been given on some of the more important categories of optimization problems that exist. The techniques used to determine solutions to optimization problems have also been mentioned. These techniques include exact and heuristic methods. The description of exact methods is out of the scope of this research. However, heuristic and metaheuristic algorithms have been explained.

This research investigates the abilities of employing both LS and SI metaheuristic algorithms in determining solutions to the ACP problems presented in chapters seven and eight. These ACP problems are single-objective *NP*-Hard optimization problems which are multi-constrained, linear and non-deterministic. They use continuous values as the decision variables to their objective functions.

CHAPTER THREE
CROP PRODUCTION AND PLANNING

3.1 Introduction

In order to present and formulate the problem of ACP, it is important to understand the crop production process. There are several stages involved in the crop production process. The decisions made at each stage will have an effect on the other stages, in sequence. Therefore, it is important that effective decisions be made at each stage of the crop production process. All decisions made will ultimately affect the overall returns gained at the end of a cropping season, and production year.

At each stage there are several factors that need to be considered. These factors (described below) are important in that they will influence the plants growth and its yield. Similarly, at each stage, there are various costs associated with the production of each crop. These accumulated costs, coupled with the potential yield and the forecasted producer prices will influence the total area of land that should be allocated for the production of each crop. Another important factor that must be considered in allocating resources is the market demand conditions of each crop. The production of each crop should not be less than what the minimum market demand is expected to be. The production should also not be more than what the maximum demand is expected to be. All these factors play important roles in making resource allocation decisions for the various competing crops that are required to be grown within a production year.

The primary factors that influence the plants growth and its yield include the climatic and soil conditions, the Crop Water Requirement (CWR) and the cultivation practices. The climatic conditions at specific geographical locations will influence the rate of evaporation from the soil surface, and the transpiration rate through the crops. The soil texture will influence the soil moisture capacity of the soil. The soils' nutritional value also plays an important role in the growth of the plant. Cultivation practices are also important throughout the plant growth stages. It involves looking after the daily needs of the plants. The factors that hinder the plants growth and affect its yield include weeds, pests, and bacteria, amongst others. These factors must be dealt with to protect the plants during their life cycles. Irrigated water applications are also important. Irrigation is important in maintaining the soils' moisture content level (Brouwer and Heibloem, 1986). The soil moisture content level should be enough to prevent the plants from wilting, and should be sufficient enough to prevent root damages. The scheduling of irrigated water is called irrigation scheduling. Irrigation scheduling is out of the scope of this dissertation.

Due to the resource limitations and the increased costs associated with crop production, it is important that effective decisions be made in managing the limited resources amongst the various competing crops that are required to be grown. The limitations of fresh water supply, and the increase in population, have resulted in an increased demand for fresh water from all sectors of industry (Schmitz *et al*., 2007). The agricultural sector has now been placed under increased pressure to used irrigated water more efficiently. This is due to the fact that the agricultural sector is mostly accused of excessive water wastage compared to other sectors of industry (Schmitz *et al.*, 2007). Therefore, it is important that optimized solutions be found concerning irrigated water allocations in crop production.

3.2 Crop Production Cycle

The crop production process is a multi-staged process. It includes crop selection, land allocations, planting, the plant growth stages, harvesting, storage and marketing (Acquaah, 2004). For each crop grown, several resources will need to be allocated for the different stages of the crop production process. The allocation of resources, at each stage, will contribute to the overall costs involved in the production of the crops. To reduce this cost, effective decisions will need to be made in resource allocation.

Majority of the costs associated with crop production include the preparation of the soil, planting, pest management, irrigated water supply and harvesting. At the level of an irrigation scheme, many tasks are done using machinery. However, there are several tasks that are required to be done by hand. Examples include the harvesting of fresh produce such as fruits and vegetables. To better understand the costs associated with agricultural production, brief descriptions are given on some of the very important stages of the crop production process.

3.2.1 Soil Preparation

The preparation of the soil takes place at the beginning of the planting season for each crop. It involves tilling the soil and using chemicals to destroy weeds, etc. Weeds need to be removed because they will compete with the crops for soil nutrition and water. The more nutrition and water used up by the weeds, the less will be available for crop development. This will influence the plants' growth and the yield obtained.

To prepare the soil, tractors are used to pull equipment for tilling the soil. An example of a piece of equipment that can be used is a plow. There are several types of plows, they include; the moldboard plow, disk plow and chisel plow. Examples of a tractor pulling a moldboard plow, disk plow and chisel plow are shown in Figures 3.2.1, 3.2.2 and 3.2.3 below.

Figure 3.2.1: Moldboard plow (www.britannica.com)

Figure 3.2.2: Disk plow (www.britannica.com)

Figure 3.2.3: Chisel plow (www.tinyfarmblog.com)

Other types of machinery include disk harrows and field cultivators. Disk harrows use steel blades to make incisions in the soil. Field cultivators are used for tillage and seedbed preparation. Examples of a disk harrow and a field cultivator are shown in Figures 3.2.4 and 3.2.5 below.

Figure 3.2.4: Disk harrow
(www.farmersguide.com)

Figure 3.2.5: Field cultivator
(www.kuhn.com)

3.2.2 Planting of Seeds

Once the soil is prepared, seeds can be sown. Seeds are sown into the soil and represent the beginning of the cropping season for each crop. The sowing of seeds can be done manually and by the use of machinery. The types of machinery used for sowing include tractors, drills and planters. Drills are used to sow seeds of crops with close spacing, such as wheat and barley. Planters create a trench, drop the seeds into it and lightly cover it with soil as they are driven through the farm plot. Examples of an agricultural drill and a planter are shown in Figures 3.2.6 and 3.2.7 below.

Figure 3.2.6: Agricultural
drill (www.tradeindia.com)

Figure 3.2.7: Agricultural planter
(www.agripak.com.pk)

3.2.3 Soil Nutrition and Pest Control

For healthy crop development, it is important to maintain a good balance of nutrients in the soil. Some of the more important nutrients that are required by the crops include nitrogen, phosphorus and potassium (Astera, 2010). If the soil lacks nutritional value then this can be improved upon by adding fertilizers such as chemical fertilizers, manure and sewage sludge. When using fertilizers it is important that the soil gets tested first. Inappropriate applications of fertilizers can cause environmental damage. The crop's development will also be affected.

The types of machinery used to apply fertilizers include tractors, planters, sprayers and spreaders, amongst others. Sprayers are used to apply chemical fertilizers to the soil and spreaders are used to apply dry fertilizers. Examples of an agricultural sprayer and spreader are shown in Figures 3.2.8 and 3.2.9 below.

Figure 3.2.8: An agricultural sprayer
(www.cropcareequipment.com)

Figure 3.2.9: An agricultural spreader
(www.gkn-walterscheild.de)

To kill pests, pesticides are used. Pests are those organisms that feed of the plants and its yield. These include insects, bacteria and mice, etc. To protect the plants from pests it is important that pesticides be used. Pesticides can also be applied using sprayers.

3.2.4 Irrigation

The difference between the CWR and the volume of rainfall that is expected to fall during the lifespan of each crop is the volume of irrigated water that is required by each crop. Irrigated water is applied at different stages during the life cycle of each crop. The application of irrigated water depends on the soil moisture content level. Irrigated water is essential for optimal plant growth. Due to irrigation, crop production is possible in areas of low rainfall.

Apart from meeting the CWR needs, irrigated water also keeps the crops cool and is used to apply liquid chemicals and safeguard against drought. Irrigated water is primarily sourced from ground water supplies, such as rivers and lakes. The transportation of irrigated water to the farm plots is done via infrastructures such as pipelines and canals. To use irrigated water, water charges need to be paid.

The primary methods used to apply irrigated water are surface (flood) irrigation, sprinkler irrigation and drip (trickle) irrigation (Brouwer *et al*., 1990).

Surface Irrigation: With surface irrigation, the water flows over the surface of the earth in furrows which are between the rows of crops. Surface irrigation is cheaper in that it does not require a lot of financial investment. However, the use of irrigated water is inefficient compared to the sprinkler and drip irrigation systems. An example of surface irrigation is shown in Figure 3.2.10 below.

Figure 3.2.10: Surface irrigation
(www.civilthought.com)

Sprinkler Irrigation: With sprinkler irrigation, water is sprayed through the air from pressurized nozzles and fall like rain drops on the crops. An example of a sprinkler irrigation system is shown in Figure 3.2.11 below.

Figure 3.2.11: Sprinkler irrigation
(www.climatetechwiki.org)

Drip Irrigation: Drip irrigation supplies water directly onto or below the soil surface. This is done through emitters that control the water flow. An example of a drip irrigation system is shown in Figure 3.2.12 below.

Figure 3.2.12: Drip irrigation
(www.turning-pro.com)

Excessive applications of irrigated water can cause damage to the root system of the plant. This will directly hinder the growth of the plant and affect its yield. Excessive irrigated water applications also cause environmental damage. These include a depletion of the source of the irrigated water, soil erosion and the washing away of fertilizers (Gajjar and Joshi, 2011). These reasons, coupled with the fact that the agricultural sector is required to use irrigated water more efficiently, make it very important that solutions be found in making efficient irrigated water allocation decisions.

3.2.5 Harvest

Depending on the type of crop, harvesting is done either by hand or machinery. Vegetables and fruits are usually harvested by hand. Examples of vegetables include tomatoes and cabbages. Examples of fruits include grapes and apples. Examples of types of crops that are harvested using machinery include maize, wheat and barley.

The types of machinery that are used for harvesting include tractors, forage harvesters and combines, etc. Forage harvesters gather, chop, and discharge forage crops as they are driven through the farm plot. Combines are used to harvest grain and seed crops. Examples of forage harvesters and combines are shown in Figures 3.2.13 and 3.2.14.

Figure 3.2.13: A forage harvester
(www.getfarming.com.au)

Figure 3.2.14: A combine
(www. deere.com)

To determine optimized solutions in managing the limited resources amongst the various crops that are required to be grown within a production year, it is important to consider the various costs associated with the crop production process.

3.3 Crop's Water Need

In making crop selection decisions it must be considered that, due to the diverse nature of plants, each plant's requirements will differ. Due to these differences, and the differences in the soil and climatic conditions at different geographical locations, the adaptability of crops at different geographical locations will differ. Therefore, in making crop selection decisions, the adaptability of the crops given the soil and climatic conditions must be considered at different geographical locations.

The soil conditions relate to the water holding capacity of the soil, its nutritional value and the transitivity of water within the soil. The transitivity factor is important for the plant's root system to absorb water. When the water is absorbed by the root system, it can then be transmitted throughout the plant (Chandy, 1993). The water will be released back into the atmosphere through the process of transpiration.

The climatic conditions relate to temperature, rainfall, humidity and wind speed, amongst others (Brouwer and Heibloem, 1986). The climatic conditions play an important role in determining the CWR of the crop's at a specific geographical location. Due to the differences in the climatic conditions, the CWR of the same crop grown at different geographical locations may differ (Brouwer and Heibloem, 1986).

Water is a major component in the physical structure of a plant. Water makes up majority of a plants' body weight (Ashraf and Majeed, 2006). For optimal physiological processes to take place within a plant, it is important that the water balance within the plant remains relatively consistent. Water that is lost through the process of transpiration must be replaced by the water absorbed through the root system of the plant. Therefore, for healthy plants and optimal yields, it is important that sufficient volumes of water be made available to the root system of a plant throughout its lifespan.

The absorption of water by the root system of the plant depends on the volume of water that has been supplied to the root surface of the plant (Chandy, 1993). Water is supplied through rainfall and irrigation. If there are inconsistencies in the application of water to the soil surface, then the soil may start to dry up. As the soil dries, the transitivity of water within the soil will decrease. This will make it more difficult for the plant's root system to absorb water.

If the water lost through transpiration is not replaced by the water absorbed through the root system of a plant, the water balance within the plant will be affected. If the plant suffers from water stress, whether mild, moderate, or severe, it will affect the process of photosynthesis, respiration, growth and reproduction within the plant (Chandy, 1993). Any water stress, particularly during the critical stages of the plant's growth, will negatively affect the plant's growth and its yield. However, some plant types are more drought resistant than others.

The main factors that influence the crop water needs of a plant include the soil factors, the climatic conditions, the crop types and the different growth stages of the plant.

3.3.1 Soil Factors

Some of the important features of the soil that are important in crop development include the soil texture, the soil moisture levels, the soil water potential and its natural or artificial drainage system.

Soil texture: The texture of the soil is its composition of sand, silt and clay (Astera, 2010). These are the particles that are found in the soil which have different sizes and feel. The percentages of sand, silt and clay in the soil will determine its field capacity. A soil type with a higher level of clay content will be able to retain more water than a soil type with a higher composition of sand. The higher the clay content in the soil, the higher will be its field capacity. Soils with higher levels of sand will have lower levels of field capacity. The field capacity refers to the maximum volume of water that the soil will be able to hold. This is the volume of water that remains after the excess volume of water has been drained from the soil (Allen *et al*., 1998). If more water is added to the soil, when it is at field capacity, the soil will not be able to retain it. The amount of time that it will take for the excess water within the soil to drain is also related to the soils' texture. The higher the field capacity of the soil, the longer it will take for excess water to be removed, and vice versa. Similar to the field capacity, the wilting point of the soil is the minimum volume of water that the soil can hold before the plant starts to wilt (Brouwer and Heibloem, 1986).

Soil Moisture: Soil moisture is the water content of the soil. If the soil moisture is below the wilting point then the plant will no longer be able to absorb water to survive. The ideal soil moisture level is when the soil moisture lies between wilting point and field capacity.

Water potential: Water potential describes the transitivity of water within the soil. This is the ability of the water to flow from one area of the soil to another. The water potential of the soil is important for the plant's root system to be able to absorb water. Water is absorbed through tiny hairs that exist on the roots of the plant (Chandy, 1993).

Natural or Artificial Drainage: Drainage is the natural or artificial removal of excess water from the soil. The removal of excess water is important in crop production. If water is left to stagnate it will cause damage to the plant's root system. This will ultimately injure the plant's development and affect its yield. The natural drainage system of some types of soil is sufficient to remove excess water. For other soil types it is important that artificial drainage systems be used (Maslov, 2009). Natural drainage occurs when there are concaved areas in the field. Any excess water will flow downwards into the concaved areas, creating ponds. One way to remove water artificially is to insert tubes into the soil. The tubes should be above the water table of the soil (Maslov, 2009). If excess water exists in the soil then it will flow into the tubes through tiny holes. The excess water can then be artificially removed. It is important not to have excess drainage. Excess drainage will remove important nutrient from the soil (Maslov, 2009).

3.3.2 Climatic Factors

The primary climatic elements that affect the crop's water need are sunshine, temperature, humidity and wind speed (Brouwer and Heibloem, 1986). Due to the climatic conditions, the evaporation and transpiration rates at one geographical location may be different from that of another. Evaporation is the removal of water vapor from the surface of the earth back into the atmosphere. Transpiration is the removal of water vapor from the stomata of the plant's back into the atmosphere. The combined removal of water through evaporation and transpiration is called evapotranspiration (Allen *et al*., 1998). The evapotranspiration rate will be higher in geographical locations that are hot and dry compared to geographical locations that are humid and cool. The wind speed also influences the crop's water need. The windier it is at a specific geographical location, the more water vapor will be released back into the atmosphere. This will increase the evapotranspiration rate. The highest crop water needs are therefore in locations that have hot, dry, windy and sunny conditions (Brouwer and Heibloem, 1986). The lowest crop water needs will be in locations that have cool, humid, cloudy and low wind speed (Brouwer and Heibloem, 1986). It is therefore observed that the crop water need of the same plant may be different from one geographical location to the next, depending on the climatic conditions. Crop's that also grow in the cooler months of the year will have lower crop water needs than those that grow in the warmer months (Brouwer and Heibloem, 1986).

3.3.3 Crop Types

Differences in the physical structure of the crop's mean that they will have different water needs. The water need of a crop such as a fully developed cotton tree will be different from the water need of a fully developed cabbage, for example. The number of days in the lifespan of each crop will also influence the water needs of the crop's. For example, the seasonal water need of a crop that grows for 90 – 100 days will be different from the water need of a crop that has a lifespan of 150 – 180 days, although their daily water needs may be the same (Brouwer and Heibloem, 1986).

3.3.4 Plant growth stages

The different growth stages in crop development relate to the volume of water that is absorbed by the crop. For example, a fully developed maize plant will absorb more water than that of a newly cultivated maize plant. This is due to the difference in their transpiration rates.

During the initial stages of crop growth, the evapotranspiration rate is mainly influenced by evaporation. This is due to the fact that the soil is more exposed to the climatic conditions because the crop, at this stage, is small. As the crop develops, it will provide more plant cover for the soil and this will reduce its evaporation rate. For fully developed crop's, the evapotranspiration rate is therefore mainly influenced by transpiration. The influence of transpiration will increase as the crop develops from its initial growth stage to the fully developed stage (Brouwer and Heibloem, 1986). Figure 3.3.1 illustrates the different growth stages of a maize plant (Brouwer and Heibloem, 1986).

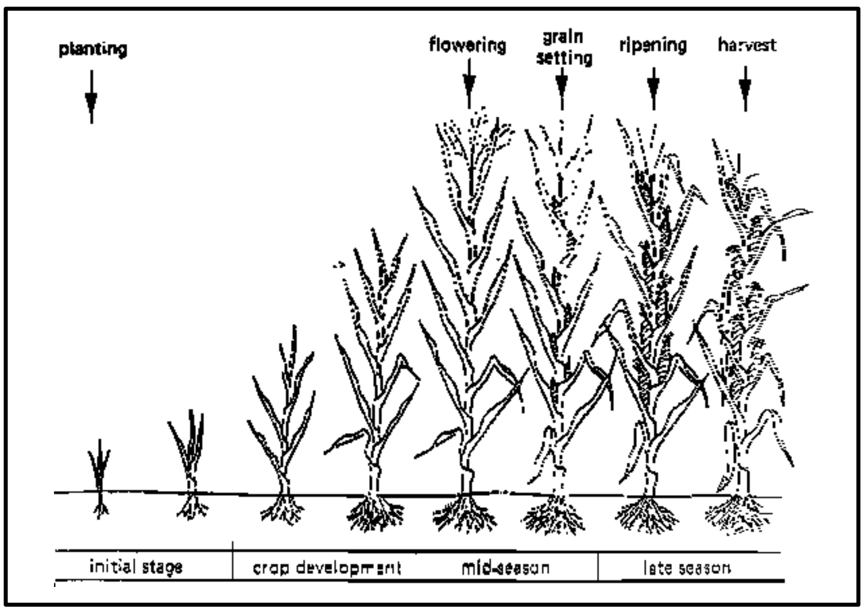

Figure 3.3.1: The different growth stages of a maize plant

The transpiration rate at the initial development stage of the maize plant is estimated to be around 50% of the transpiration rate of a fully developed maize plant. The fully developed maize plant is found in the mid-season stage. This is when the transpiration rate is 100%. During the crop's development stage, the crop's water need will gradually increase from 50% to the 100% level in the mid-season stage. The crop's water need in the late season stage will differ depending on the crop type. For freshly harvested produce, such as lettuce, the crop's water need in the late season stage will remain the same as in the mid-season stage. This is because the crops would need to be harvested fresh. Therefore, the crop's water need must remain the same until the day of harvest. For dry harvested crops such as cotton, maize and sunflower the crops are allowed to dry out in the late season stage. This is where their water needs will be the least (Brouwer and Heibloem, 1986).

3.4 Demand and Supply Conditions

The resulting market prices of produce are dependent on the demand and supply conditions within the deregulated market environment. The price of a harvest is settled when the producer and purchaser agree upon a selling price. The producer will want to maximize the profits earned while the purchaser will want to purchase the produce at the lowest possible price. The price that is agreed upon by both parties is called the equilibrium price (Whelan and Msefer, 1996). The derivation of the equilibrium price is illustrated in Figure 3.4.1.

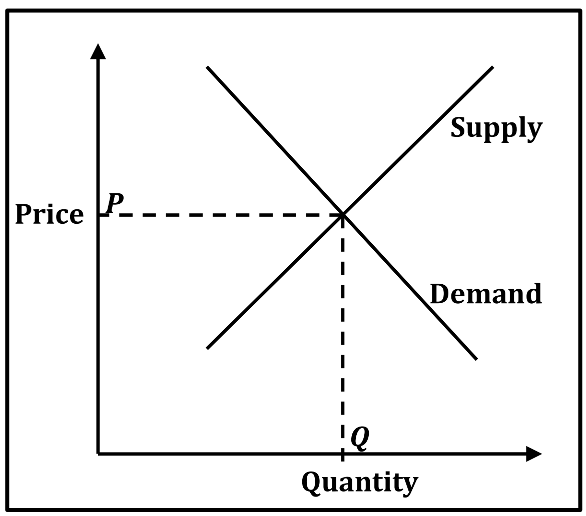

Figure 3.4.1: The derivation of the equilibrium market price.

In Figure 3.4.1, P represents the price of the produce, while Q represents the quantity demanded. The point at which both the producer and purchaser agree upon a selling price is the point at which quantity Q will be traded at a price P. At this point the demand and supply will be in equilibrium. At any price below P, the quantity of produce demanded will increase. This is due to the desire of the purchaser to buy at a lower price. At any price above P, the demand will decrease due to the reluctance of the purchaser to pay a higher price.

If there is a shortage of produce in the market, the producers will sell at higher prices. If the purchasers need the produce then they will be forced to pay these higher prices. If there is surplus produce in the market then the producers will have to sell at lower prices. This is due to the existence of competition among producers trying to sell their produce. They would need to sell their produce so that they don't incure any losses.

Another factor that influences the market demand and supply conditions is the consumer preferences (Lovewell, 2012). If there is an increase in comsumer preference, then the demand for a produce will increase. Similarly, if there is a decrease in consumer preference, then the demand for a produce will decrease. Other factors that influence the market demand and supply conditions include the weather, technology and the cost of transportation, amongst others (Lovewell, 2012).

3.5 Case Studies

The case studies used in this research are the Vaalharts and Taung Irrigation Schemes. These are neighboring irrigation schemes. Their location is situated on the area bordering the Northern Cape and North West Province in South Africa (Grove, 2008).

The Vaalharts Irrigation Scheme (VIS) is the largest irrigation scheme in South Africa and one of the largest irrigation schemes in the world. The VIS covers an area of around 36,950 hectares of prime agricultural land (Grove, 2008). Situated near the VIS is the Vaal River. The irrigated water that is currently supplied to the VIS is extracted from the Vaal River and is supplied to the farm plots via the Vaalharts Canal System (Grove, 2008). Artificial drainage systems are also in place. It extracts the excess water into the Harts River, which is west of the scheme.

The Taung Irrigation Scheme (TIS) is situated north of the VIS. TIS consists of a total of 3,764 hectares of agricultural land (Smook *et al*., 2008). The irrigated water that is supplied to the TIS is also currently supplied via the Vaalharts Canal System, although the Taung Dam is situated nearby.

Figure 3.5.1 shows a satellite image of the neighboring irrigation schemes. The figure specifies the location of each irrigation scheme. It also specifies the location of the Taung Dam and the Vaal River.

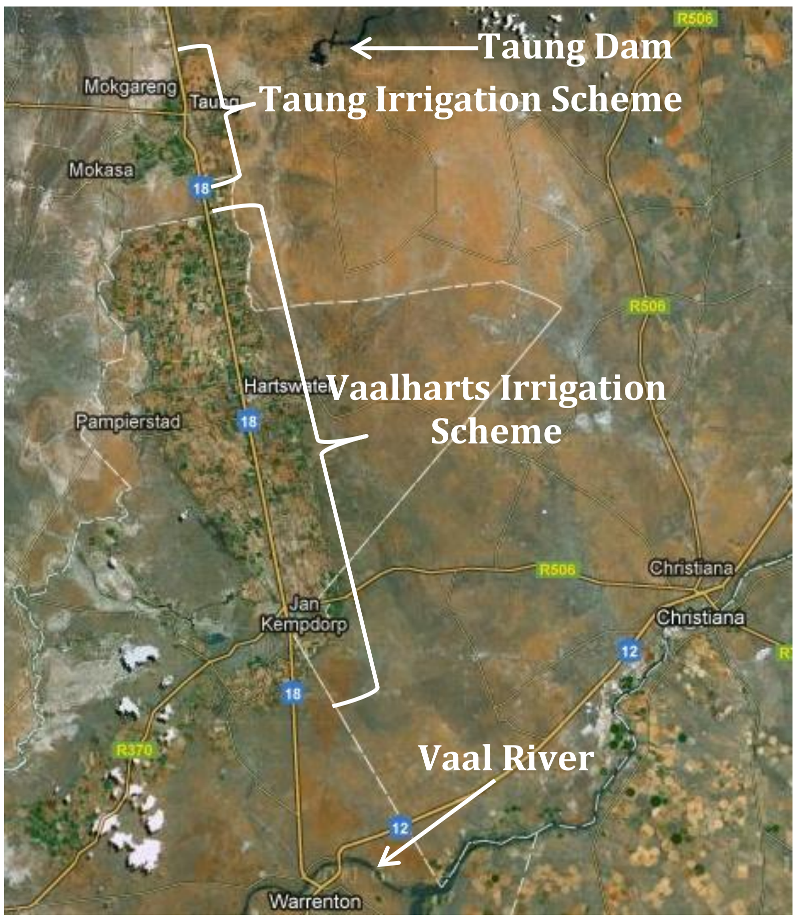

Figure 3.5.1: Location of the Vaalharts Irrigation Scheme, Taung Irrigation Scheme, Vaal River and Taung Dam

To better understand crop production at both the VIS and TIS, brief descriptions are given concerning the climatic and soil conditions, the irrigated water supply and the crop preferences in the area.

3.5.1 Climatic Conditions

This area is known for its very warm summers and very cold winters. Due to the very cold winters, frost occurs. The average rainfall in the area averages at around 440 millimeters (mm) per annum. Apart from the volume of rainfall being low, it is also very irregular (Maisela, 2010). Rain primarily falls between the months of November through to April. It is at its lowest between the months of March and October. Due to the low volume of rainfall and the irregular rainfall patterns, it is necessary that irrigated water be supplied to the area to facilitate crop production.

The highest temperatures occur between the months of November and February. The maximum temperatures for these months average over 30°C. The minimum temperatures for these months average at around 15°C. The temperature is at its lowest in the months of June and July. The maximum temperatures in these months average around 19°C and the minimum temperatures average at around 2°C (Maisela, 2010).

Table 3.5.1 below shows the statistics for the average temperature and rainfall patterns that have been determined over a period of 36 years (Maisela, 2010).

Table 3.5.1: Mean temperature and rainfall statistics as determined over a 36 year period.

	Jan	**Feb**	**Mar**	**Apr**	**May**	**Jun**	**Jul**	**Aug**	**Sep**	**Oct**	**Nov**	**Dec**
Mean Temp	24.8	23.8	21.7	17.9	13.7	10.5	10.5	12.8	16.9	19.8	22.0	23.9
Mean Rainfall	75.9	63.5	71.8	51.6	19.9	9.5	4.3	8.6	11.3	24.6	45.7	58.0

Figures 3.5.2 and 3.5.3 give graphical representations of the mean temperatures and rainfall statistics as given in Table 3.5.1.

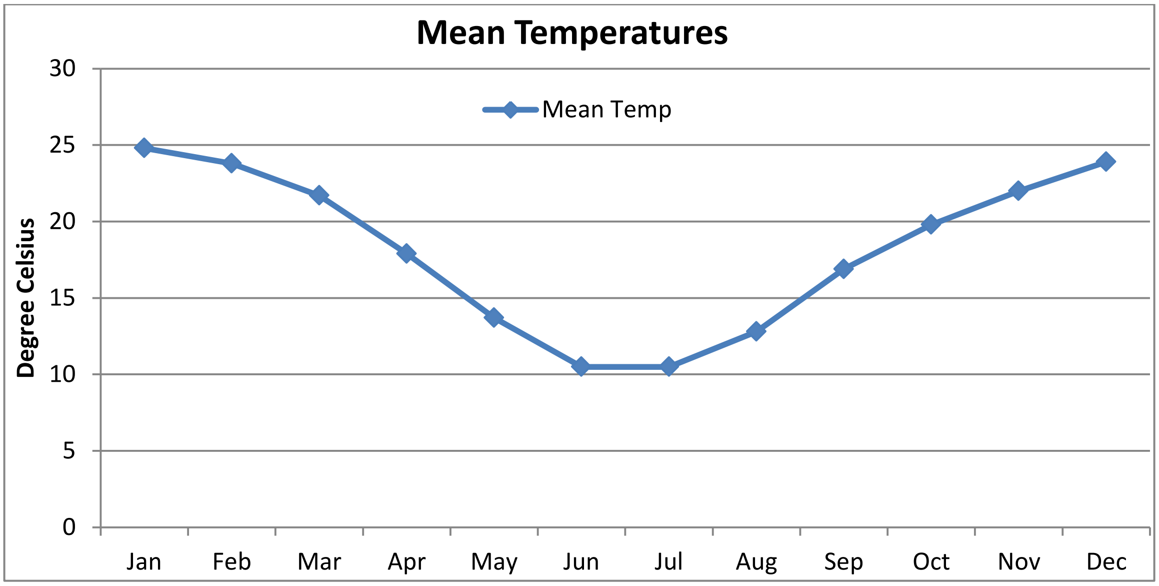

Figure 3.5.2: Mean temperature statistics as determined over a period of 36 years

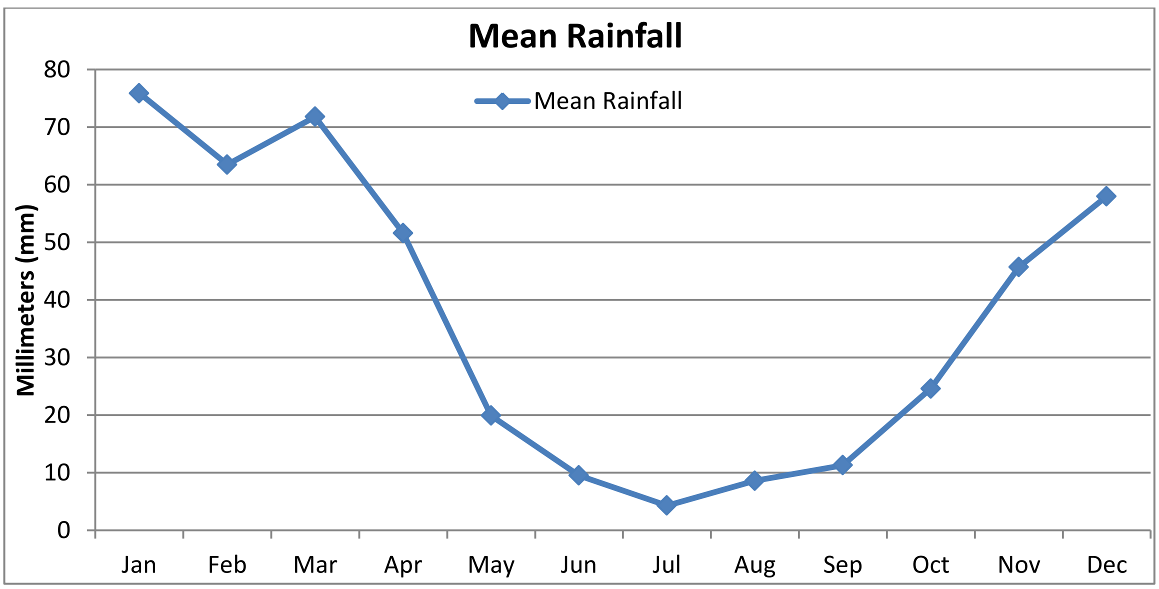

Figure 3.5.3: Mean rainfall statistics as determined over a period of 36 years

3.5.2 Soil Characteristics

The two main types of soils found in this region are Hutton (Mangano) and Clovelly (Cunbury) (Grove, 2008; Maisela, 2010). The soil texture consists of around "*8% clay, 2% silt, 68% fine sand and 22% medium and course sand*" (Maisela, 2010). The high percentage of sand in the soil means that it has a relatively low field capacity, low water holding capacity and low level of fertility. With rainfall and irrigation, the soil will get compacted and this strains the plant's root system in terms of its development. The depth of the soil ranges from around 0.9 meters to 1.8 meters (Grove, 2008).

3.5.3 Irrigated Water Supply

Irrigated water is supplied to the irrigation schemes via the Vaalharts Canal System. The two primary canals that transport the water to the schemes are the North and the West canals. These canals supply irrigated water to a system of feeder canals, which in turn supply irrigated water to the community canals. The community canals supply the irrigated water to the farm plots (Grove, 2008). Irrigated water is supplied at a quota of 9,140 m^3 ha^{-1} $annum^{-1}$ to the farm plots at the VIS. It is supplied at a quota of 8,140 m^3 ha^{-1} $annum^{-1}$ to the farm plots at the TIS. A water charge of 8.77 cents m^{-3} needs to be paid to the Water User Association (WUA).

It is estimated that around 62% of all fresh water supply in South Africa is used by the agricultural sector (Oelofse and Strydom, 2010). To reduce irrigated water wastage within the agricultural sector itself water charges are employed. To reduce the cost of irrigated water, farmers are therefore required to use irrigated water more efficiently. The aim of implementing water charges is to conserve water.

3.5.4 Crop Preferences

A list of some of the most adaptable crops at the VIS and TIS is presented in Table 3.5.2 below (Maisela, 2010).

Table 3.5.2: Some of the adaptable crops in the area.

Crop Types	Crops	Well Adapted/Adaptable
Summer Crops	Cotton	Well Adapted
	Maize	Well Adapted
	Groundnuts	Well Adapted
	Tomatoes	Well Adapted
	Pumpkins	Well Adapted
	Dry Beans	Adaptable
	Soya Beans	Adaptable
Winter Crops	Wheat	Well Adapted
	Barley	Well Adapted
	Canola	Adaptable
	Cabbage	Well Adapted
	Onions	Well Adapted
Perennial Crops	Lucerne	Adaptable
	Pecan Nuts	Well Adapted
	Olives	Well Adapted
	Citrus	Adaptable
	Wine Grapes	Adaptable

The cash crops that are the most important in this area include maize, wheat, barley, lucerne and ground nuts (Grove, 2008). Maize and ground nuts are summer crops. They are usually grown in sequence with wheat and barley, which are winter crops. Lucerne is a perennial crop which grows all year around. The forecasted producer prices and the crop yields play important roles in determining the area of land that should be allocated for the production of each crop.

3.6 Previous Research

Previous studies in crop and irrigation planning have used both single and multi-objective mathematical models. Many optimization techniques have been used to provide solutions to these models. These include; Linear Programming (LP), Dynamic Programming (DP), Simulated Annealing (SA), Evolutionary Algorithms (EAs) and Particle Swarm Optimization (PSO), amongst others.

Mohamad and Said (2011) proposed a crop-mix planning model. The model takes into consideration limited resources such as finances and acreage. The research used LP to determine the optimal solution which maximized the returns gained. Sunantara and Rimirez (1997), used DP to solve a problem of irrigated water allocation and scheduling using a two-stage decomposition approach. The first stage solved the problem of seasonal water and acreage allocation. The second stage solved the problem of daily water scheduling as a function of the root-zone soil moisture content levels. Wardlaw and Bhaktikul (2004) used the GA to solve a problem of irrigated water scheduling, using a 0-1 approach. They found that the GA performed well, by being able to distribute irrigated water to several farm plots in satisfying the soil moisture content levels under water stress conditions. The water allocations were done on a rotational basis. Georgiou and Papamichail (2008) used SA in combination with the Stochastic Gradient Descent Algorithm to determine solutions concerning the optimized water release policies of a reservoir. The released water needed to be allocated efficiently amongst the various crops being grown. To maximize profits, an optimized cropping pattern needed to be determined.

Sarker and Ray (2009) proposed an improved EA known as the Multi-objective Constrained Algorithm (MCA). MCA was used to provide solutions to a multi-objective crop planning problem. The research found that MCA performed relatively better compared to the other two optimization techniques used. These techniques included the ε-constrained method and the Non-dominated Sorting Genetic Algorithm (NSGAII). Adeyemo and Otieno (2010a) compared two different versions of their Multi-objective Differential Evolution Algorithm (MDEA) in determining solutions that tried to maximize the potential irrigation benefits that could be achieved at the Vanderkloof dam, in South Africa. The two different versions were called MDEA1 and MDEA3. These versions differed in the crossover techniques used. Adeyemo *et al* (2010b) used DE to determine improved solutions in using single objective crop planning models, compared to the solutions determined in using MDEA to provide solutions to the same problem formulated as a multi-objective crop planning model. Pant *et al* (2008) employed the DE algorithm to provide solutions to a crop planning problem under adequate, normal and limited irrigated water supply. The objective was to maximize the net benefits gained, under these conditions. It was found that the DE performed better than the programming tool LINGO. Pant *et al* (2009) investigated the performances of four EAs in providing solutions to a crop planning problem. These algorithms included the GA, PSO, DE and Evolutionary Programming (EP). Solutions were also determined using LINGO. The solutions found showed that, from all heuristic algorithms, GA performed poorly and that DE, PSO and EP were all comparable. Raju and Kumar (2004) compared the performances of GA and LP in providing solutions to a crop planning problem. The objective was to maximize the net benefits gained. The performances of GA and LP were relatively close. It was concluded that GA is an effective metaheuristic algorithm that can be used in irrigation planning. Reddy and Kumar (2007) studied the effectiveness of using Elitism-Mutation Particle Swarm Optimization (EMPSO) in determining the short-term release policies of irrigated water from a reservoir in water scarce conditions. The study concluded that the heuristic algorithm is effective in providing short-term solutions for multi-crop irrigation.

3.7 Conclusion

This chapter reviews the several stages involved in crop production. The different stages include crop selection, land allocation, planting, plant growth stages, harvesting, storage and marketing. At each stage, the decisions made are important and will directly influence the success of crop production at an irrigation scheme.

In making resource allocation decisions during the crop production process, the costs associated with crop production must be taken into account. There are several costs that must be considered. These include the costs associated with the preparation of the soil, planting, pest management, irrigated water supply and harvesting, amongst others. To determine optimized solutions in crop planning, the various costs associated with crop production must be taken into account.

Another factor that must be taken into account in making crop planning decisions is the geographical location of the irrigation scheme. Due to differences in the nature of the crops, and their suitability to different soil and climatic conditions, the adaptability of crops will differ from one geographical location to the other. The crops selected to be produced must be adaptable to the given environmental conditions at a specific geographical location. In making land allocation decisions, the expected yield, forecasted producer prices and the demand conditions of each crop will influence the area of land that should be allocated for the production of each crop.

The limited supply of fresh water, and the increase in population, has also resulted in an increase in the demand for fresh water supply from all other sectors of industry. Due to this increased demand, the agricultural sector has been placed under increased pressure to use irrigated water more efficiently, making it essential that optimized solutions be found concerning irrigated water allocations amongst the various competing crops that are required to be grown.

This chapter also presents the case studies addressed in this research. This research addresses the ACP problems at the Vaalharts and Taung Irrigation Schemes. These irrigation schemes are neighboring irrigation schemes, which are situated on the border separating the Northern Cape and the North West Province in South Africa. Descriptions of the environmental conditions and a list of some of the adaptable crops in the area have also been given.

Finally, previous researches in crop and irrigation planning have been discussed.

CHAPTER FOUR
SPACE ALLOCATION IN ANNUAL CROP PLANNING

4.1 Introduction

Space allocation is a very important managerial responsibility. It involves allocating a limited area of available space amongst the various demanding entities that require space utilization (Adewumi and Ali, 2010; Silva, 2003). The objective in making space allocation decisions is to maximize the amount of satisfaction that is given to each demanding entity, in optimizing the problems' objective. In ACP, this involves allocating a limited area of agricultural land amongst the various competing crops that are required to be grown within a production year. The objective is to optimize the returns received. Any mismanagement in the way space allocation decisions are made will negatively affect overall operational costs (Silva, 2003).

Determining optimized space allocation decisions in crop planning is very difficult. There are several uncertain factors that must be taken into consideration in making decisions. Some uncertain factors include the climatic conditions, the crop yields, the fluctuating market prices and the demand and supply conditions. Other factors that will influence space allocation decisions include production costs, cropping patterns, planting schedules, harvesting schedules and the farm plots sizes, amongst others. All these factors will influence the land allocation decisions made.

Despite the difficulty in determining optimized hectare allocations, many producers still rely on manual methods to make space and resource allocation decisions. The inefficient use of limited resources will affect the production costs and the returns gained. For optimized solutions to be found in making space and resource allocation decisions, it is important that information and technology be combined to determine solutions.

This chapter describes the ACP problem as a Space Allocation Problem (SAP). Descriptions of the several complexities involved with determining feasible solutions are given. Many of the hard and soft constraints that need to be satisfied, in determining feasible solutions are listed. The method used to mathematically formulate the ACP problem as a SAP is also given.

4.2 Space Allocation in Crop Planning

SAP's are very difficult optimization problems in literature. Examples of SAP's include the space allocation at tertiary institutions (Silva, 2003; Adewumi and Ali, 2010) and the shelve space allocation problem at the level of supermarkets (Bai, 2005). Space allocation is a complex problem that involves allocating a limited area of available space amongst a set of demanding entities that require space utilization (Silva, 2003). The objective in allocating the space is to grant as much satisfaction as possible to all demanding entities involved in optimizing the problems' objective. In determining feasible solutions to multi-constrained SAP's, several hard and soft constraints will need to be satisfied. The mathematical formulation of the problem and the types of constraints that will need to be satisfied are problem specific.

In this research, the allocation of a limited area of agricultural land amongst the various competing crops that are required to be grown within a production year is viewed as a SAP. In allocating land amongst the various crops, this research considers the irrigated water requirements and the variable costs associated with the crop production process. The optimized solutions found must allocate the limited area of agricultural land amongst the various competing crops in a way that will optimize the irrigated water requirements and the variable costs associated with the production of each crop. The objective will be to maximize the total gross profits earned. Feasible solutions must satisfy the minimum and maximum market demand constraints. To determine feasible solutions, the farm plot sizes and multi-cropping practices must be considered.

A farm plot is an area of agricultural land that is allocated for crop production. Farm plots are categorized by the number of different crops that are grown in sequence on it within the year. Single-crop farm plots are areas of land that have been allocated for the production of perennial crops. Perennial crops grow all year around on the single-crop farm plots. Perennial crops are harvested once or several times within a year, depending on the crop. Examples of perennial crops include fruit trees and lucerne. Fruit trees are usually harvested once a year. Lucerne is harvested several times within the year.

Multi-cropping is a cultivation practice that involves growing different crops on the same farm plot within a year (Charles, 1986). The types of multi-cropping techniques used include sequential cropping and inter-cropping. Sequential cropping is when selected crops are allowed to be grown in sequence of one another on the same farm plot within a year. Inter-cropping is when different types of crops grow together on the same farm plot within a year. The multi-cropping practice investigated in this research is sequential cropping. All references made to multi-cropping in this research work relate to sequential cropping.

The farm plots allocated for multi-cropping include the double-crop, triple-crop and quadruple-crop plots, etc. (Grove, 2008). Double-crop farm plots are used to cultivate two different crops which are grown in sequence within a year. These may include certain seasonal crops such as the summer and winter crops. An example of double-cropping is the cultivation of maize and wheat. In South Africa, maize is a summer crop and wheat is a winter crop. These crops are usually grown in sequence on a double-crop farm plot within the year. Triple-crop plots are used to cultivate three different types of crops which are grown in sequence on a triple-crop farm plot within a year, and so on.

For multi-cropping to be successful, the crops selected to be grown in sequence need to be selected carefully. The planting and harvesting schedules of the crops grown in sequence must not conflict and should be beneficial to each other. There are several benefits to multi-cropping, if done correctly. Some of the advantages of multi-cropping include (Charles, 1986):

- It produces higher returns from a farm plot as multiple crops can be grown on the same plot of land within a year.
- It helps protect against drought, pests, diseases and weed developments. This will reduce the costs of fertilizers and pesticides, etc.
- Nutritional value gets added back to the soil.

With the existing irrigation schemes, the total area of land allocated for the different farm plot types generally remain the same.

Formulating a crop planning problem as a SAP involves taking into consideration the limited area of agricultural land available for crop production on each type of farm plot. Once it has been decided which crops will be grown in sequence on a farm plot, solutions will need to be determined concerning land allocation amongst the various competing crops that are required to be grown. The objective in making land allocation decisions will be to maximize the outputs obtained from it. For solutions to be feasible, there are several constraints which are associated with multi-cropping that would need to be satisfied. Space allocation in crop planning is an NP-Hard type optimization problem in agricultural planning.

Despite the difficulties associated with determining optimized solutions in crop planning, many crop producers still employ traditional methods in making resource allocation decisions. Inefficiencies in making resource allocation decisions will affect the overall outcomes determined at the end of the cropping season, and production year. To determine optimized solutions in crop planning, it is important that information and technology be combined in determining solutions. The information that is needed includes having knowledge of the potential crop yields, the forecasted producer prices, the demand and supply conditions, the climatic conditions, the CWR’s and the variable costs associated with crop production, amongst others. The information required should preferably be location specific. This information can be determined from local farmers in them observing the crop yields, production costs, market prices and market demand and supply conditions from previous years. Information can also be determined from agricultural advisory services (Kantanantha, 2007). Information concerning the climatic conditions can be determined from the local weather stations. For the seasonal CWR needs, information can be determined from farmers, advisory services or from publications in literature.

The mathematical models commonly used to formulate SAP's include bin-packing, assignment modeling and knapsack modeling (Silva, 2003). This research employs an adapted knapsack model in formulating the mathematical models for the ACP problems for new and existing irrigation schemes.

4.3 Knapsack Modeling

Knapsack models are known *NP*-Hard optimization models (Pisinger, 1995). It involves assigning a subset of items, each of which has an associated profit and weight value, into a knapsack or knapsacks. The objective in trying to fill the knapsack(s) is to maximize the total accumulated profits of all the items selected. The accumulated weight of all the items selected must not exceed the maximum capacity of the knapsack(s). Figure 4.3.1 illustrates the objective of a knapsack problem with a simple example.

In Figure 4.3.1, a knapsack (bag with shoulder straps) has a maximum capacity of 15 kilograms (kg). The problem is to determine a solution that will fill the knapsack with a subset of the items in a way that will maximize profit, without exceeding the maximum capacity of the knapsack.

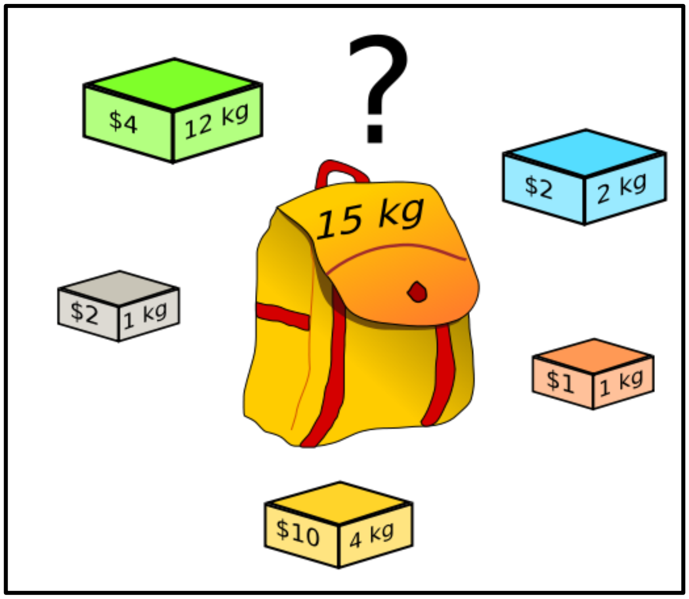

Figure 4.3.1: Illustration of a one-dimensional knapsack problem (http://en.wikipedia.org/wiki/Knapsack_problem)

Several knapsack models exist in literature. The differences in the models lie in the way that the items are allowed to be selected, and the number of knapsacks involved (Nyonyi, 2010). The types of knapsack models that exist include binary, fractional, bounded and multiple knapsack models, amongst others. The illustration of the problem given in Figure 4.3.1 can be formulated mathematically using a binary knapsack model if each item is allowed to be selected at most once. It can be formulated using a fractional knapsack model if fractions of the items are allowed to be selected. It can be formulated using a bounded knapsack model if there are bounds that exist in selecting the items. If there are multiple knapsacks involved, the problem can be formulated using a multiple knapsack model.

Adaptations of knapsack modeling are possible. The way a model is adapted will depend on the type of problem addressed. For example, if the problem involves multiple knapsacks with each item only allowed to be selected at most once then the adapted model used will be a binary-multiple knapsack model. This research uses an adapted knapsack model to formulate the ACP problems for new and existing irrigation schemes. The adapted model used is a bounded-fractional-multiple knapsack model, with an additional constraint. The additional constraint is that the total capacity of all the items in each of the knapsacks must not exceed the total summation of the maximum capacities of all the knapsacks added together.

4.4 Bounded-Fractional-Multiple Knapsack Modeling

The aim of this bounded-fractional-multiple knapsack model is to fill each knapsack with items from allocated subsets. The objective is to maximize the total accumulated profits while satisfying the constraints. For each subset of items, fractions of the items are allowed to be selected and each item must be selected once. An additional constraint is added such that the summation of the total weight values of all the items selected must not exceed the total sum of the maximum capacity allowed in each knapsack. The formulation of the bounded-fractional-multiple knapsack model, with an added constraint, is as follows:

Suppose there are a total of m knapsacks of capacity k_j, i.e. k_j, $\forall\, j = 1, \dots, m$. For each knapsack k_j suppose that x_{ij} ($\forall\, i = 1, \dots, n_j$) is used to fill k_j. Each x_{ij} has an associated profit p_{ij} and weight w_{ij} value. Each x_{ij} is allowed to contribute a fraction f_{ij} of itself ($0 < f_{ij} \leq 1$) into the knapsack k_j. The fraction of x_{ij} must fall within the lower (Lb_{ij}) and upper bound (Ub_{ij}) values of each x_{ij}. The total weight of the maximum capacities of all knapsacks is T. The mathematical model is as follows;

Maximize:

$$f(x) = \sum_j^m \sum_i^{n_j} p_{ij} x_{ij} \tag{4.1}$$

Subject to:

$$\sum_{i=1}^{n_j} w_{ij} x_{ij} \leq k_j, \qquad \text{for } j = 1, 2, \dots, m \tag{4.2}$$

$$x_{ij} = \begin{cases} 1 & \text{if item } x_{ij} \text{ gets put into knapsack } k_j \\ 0 & \text{otherwise} \end{cases} \tag{4.3}$$

$$0 < f_{ij} \leq 1 \tag{4.4}$$

$$Lb_{ij} \leq f_{ij}x_{ij} \leq Ub_{ij} \tag{4.5}$$

$$\sum_{j=1}^{m} \sum_{i}^{n_j} w_{ij}x_{ij} \leq T \tag{4.6}$$

In relation to crop planning, p_{ij} represent the profits earned. w_{ij} represents the allocation of land for each crop x_{ij}. The ACP mathematical models introduced in this dissertation are based on this mathematical formulation.

4.5 Problem Description

Irrigation schemes commercially produce several types of crops for both the local and international markets. The production of crops is subject to resource limitations, and the supply of crops to the markets should be within the markets' demand. The objective in producing crops, given the limited resources and the demand and supply conditions, is to maximize the total gross profits that can be earned. In the sale of the harvests, the producers need to consider the costs that were involved in the crop production process. The costs involved with crop production include:

1. **Labour** – Depending on the particular crop, and the area of land allocated for production, the producer may allocate few to many labourers.
2. **Materials** – These include pesticides and fertilizers, amongst others
3. **Transportation costs** – Specialized vehicles are used in planting, fertilizing, harvesting and transporting of the harvests.
4. **Water costs** – This is the cost of irrigated water. Irrigated water needs to be applied according to the CWR of each plant. Irrigated water must be scheduled to maintain the soil moisture content balance within the soil.
5. **Others** – Other types of costs include the cost of seeds, electricity, household expenses and storage costs.

In determining solutions to the ACP problem, these costs are considered constant values. They can be determined by monitoring the crop production costs from previous years. It can also be determined from published statistical reports on crop production. These reports may include provincial and national government reports.

The purpose of developing mathematical models is to help decision makers make effective decisions in crop planning when trying to answer the following questions:

1. Which crops should be selected for cultivation?
2. What is the area of land that should be allocated for the production of each crop within a production year?
3. Which crops should be selected to be grown in sequence on the same farm plot?
4. What is the irrigated water requirement of each crop, given the area of land that should be allocated for its production?
5. What is the cost associated with the production of each crop, given the area of land that should be allocated for its production?
6. What is the total gross profit earned in producing each crop, given the expected crop yields, the forecasted producer prices and the area of land allocated for its production?
7. What will be the overall gross profit earned in producing all crops within a production year?

ACP for new and existing irrigation schemes involves allocating a limited area of agricultural land amongst the various competing crops that are required to be grown within a production year. The objective in allocating land is to optimize the limited resources available for crop production. The limited resources include the land itself, irrigated water and the variable costs associated with crop production. Solutions in making resource allocation decisions will need to be determined for each crop being cultivated.

To determine feasible solutions to the ACP problem, several hard and soft constraints will need to be satisfied. However, the requirements in determining solutions to the ACP problem are as follows:

1. The crops to be grown and the farm plot requirement for each crop must be known.
2. The total area of land available for crop production and the total area of land available for each farm plot must be known.
3. Information about the market demand conditions must be available.
4. For each crop type, the expected crop yield must be known.
5. The volume of irrigated water that can be supplied to the farm plots must be known, as well as the cost of this irrigated water. The rainfall pattern also needs to be known. This is used to determine the optimized irrigated water allocations of each crop.
6. The variable costs associated with the production of each crop also need to be determined.

Crop selection is the first step in the crop production process. It is a separate stage from the land allocation stage. Crop selection requires consideration of the market demands, the expected yields, the forecasted producer prices of each crop, the adaptability of each crop to the environmental conditions and the variable costs associated with the production of each crop. Crop selection is out of the scope of this research. In this research, it is assumed that the crops selected to be cultivated within a production year has already been selected. Once the crop selection is finalized, solutions will need to be determined concerning resource allocations amongst the competing crops.

Determining land allocations solutions for an existing irrigation scheme requires knowledge of the total area of land available for crop production. The total area of land for each farm plot type must also be known. The farm plot types include the single-crop, double-crop and triple-crop plots, etc. The land allocated to each farm plot type at existing irrigation schemes is usually fixed. This is primarily due to multi-cropping practices. The purpose of formulating a mathematical model for an existing irrigation scheme is to determine the area of land that should be allocated for the production of each crop. The land allocation must be done while trying to optimize resource allocations amongst the various competing crops that are required to be grown. In this research, the resources that would need to be optimized include the limited area of agricultural land, irrigated water requirements and the variable costs associated with the production of each crop. The feasible solutions found must not break the multiple land and irrigated water allocation constraints.

The minimum and maximum market demand for each crop should be determined. The minimum supply will ensure that the minimum market requirements are met. A constraint on the maximum supply will ensure that an excess amount of crop yield is not produced. The feasible solutions determined in making land allocation decisions need to consider these constraints. The minimum and maximum market demand conditions are location specific factors.

The probable crop yield of each crop to be grown must be determined. The yield and the forecasted producer prices are important factors in making resource allocation decisions. These factors will directly affect the resource allocation decisions made and the total gross profits earned at the end of the cropping season for each crop, and at the end of the production year for all crops.

To determine optimized irrigated water allocation for each crop, the rainfall pattern must be considered. The difference between the CWR of each crop and the volume of rainfall that is expected to fall during its lifespan is the volume of irrigated water that is required by each crop. Determining optimized solutions to irrigated water allocation is important. Excessive applications of irrigated water can also cause environmental damage. To control the usage of irrigated water, crop producers are required to pay water charges meaning that they need to produce more output per meter cubed (m^{-3}) of irrigated water used.

The variable costs associated with crop production also need to be optimized in determining resource allocation solutions. The variable costs of production are the accumulated costs of the various inputs associated with the production of each crop. These costs include the cost of labor, fertilizers, fuel, electricity, storage costs, etc. It is up to the crop planner to determine the variable cost of production value of each crop. The production costs will differ for each crop produced. It will also be different at different geographical locations.

Once this information is known, solutions can be determined by formulating mathematical models. The purpose of formulating the models is to determine optimized resource allocations amongst the various competing crops to be grown. The mathematical models developed in this research consider the above mentioned factors in determining resource allocations. The objective is to determine the land allocations amongst the various competing crops in a way that will optimize the irrigated water requirements and the variable costs associated with the production of each crop. The feasible solutions found must satisfy the market demand conditions for each crop and the multi-cropping practices.

There are also multiple hard and soft constraints, and objectives, that are associated with determining feasible solutions to the ACP problem. Some of these objectives and constraints include:

1. Optimize the total gross profits that can be earned within a production year, in producing all the crops.
2. Optimize the resource allocations amongst the various competing crops. The resource allocations include the limited area of agricultural land, the irrigated water requirements and the variable costs associated with the production of each crop.
3. Maximize the available space given to each crop being produced.
4. Satisfy the market demands.
5. The allocation of land amongst the various competing crops should be done as fairly as possible.

Some of the hard and soft constraints associated with determining feasible solutions are as follows:

Hard constraints:

1. Perennial crops must only be allocated on the single-crop farm plots. Only two crop groups are allowed to be grown in sequence on the double-crop plots. Only three crop groups are allowed to be grown in sequence on the triple-crop plots, and so on.
2. The crops that have been selected to grow in sequence on the same farm plot must not have conflicting planting and harvesting schedules.
3. The total area of land that has been allocated to each crop, which belongs to a particular crop group, must be less than or equal to the total area of land that is available for crop production for that particular crop group.
4. The total area of land that has been allocated to each crop group, which has been allocated the most area of land on their respective farm plots, must be less than or equal to the total area of land that is available for crop production on an irrigation scheme.
5. The total volume of irrigated water that is required to meet the CWR's of all crops must not exceed the total volume of irrigated water that can be supplied to the irrigation scheme, within a year.
6. The area allocations given to each crop should produce yield that would satisfy the market demand conditions.
7. Every crop that is required to be grown must be allocated a portion of land.

Soft constraints:

1. Allocate as much land as possible to each crop in determining optimized solutions.
2. Minimize the total irrigated water requirements and the variable costs associated with the production of each crop type.

The feasible solutions found must not break any hard constraints and should satisfy as many soft constraints and objectives as possible.

4.6 Research Assumptions

This research makes the following assumptions:

1. The methods introduced in this research will be adopted at the beginning of a crop production year, for new and existing irrigation schemes.
2. The total area of land available for agricultural production is known.
3. For an existing irrigation scheme, the total area of land that is available for each farm plot type remains fixed.
4. Multiple crops are required to be grown.
5. The crops that are to be grown should have already been selected.
6. The forecasted producer prices for each crop should have been determined. This information can be determined from published literature (Kantanantha, 2007), by observing the market prices from the previous production years or consulting advisory services, amongst others. It is important to get the forecasted producer prices that are relevant to the markets in which the harvests will be sold.
7. No restrictions are placed on the availability of the inputs associated with crop production.
8. The variable costs of production of each crop can be calculated.
9. The CWR of each crop is known.
10. It is acceptable to use the average rainfall pattern in determining optimized irrigated water requirements.
11. The crop's yield under optimal cropping practices must be known.
12. It is assumed that the results are determined under optimal cropping practices and favourable conditions. No unforeseen circumstances such as natural disasters are considered. This includes drought and flooding.
13. The crops will be planted and harvested as scheduled.

It is preferable that the information required is determined locally. The market conditions of one geographical location will differ from the next. The government also publishes national reports on the statistics of the major crops produced annually. For each crop produced, these reports may give the yields obtained and the market prices. These reports indicate the conditions of the markets as a whole throughout the country, and are not location specific (Kantanantha, 2007). These reports are acceptable and can be used as benchmark data. However, it is preferred that location specific data be used. In this research, national government reports have been used.

4.7 Conclusion

This chapter describes ACP as a SAP. SAP's are amongst the most difficult optimization problems in literature. The formulation of the problem requires taking into consideration several factors that will affect the ACP decisions made. Amongst these are the climatic conditions, crop yields, fluctuating market prices, the demand and supply conditions, the production costs, the cropping patterns and the planting and harvesting schedules. There are several hard and soft constraints that would need to be satisfied in order to determine feasible solutions. The purpose of formulating the ACP mathematical models is to factor in all the relevant information associated with determining feasible solutions, while minimizing the number of decision variables used.

The formulation of the ACP mathematical models presented in chapters seven and eight are based on an adapted knapsack model. This model is a bounded-fractional-multiple knapsack model, which has been discussed and presented in this chapter. Knapsack modeled optimization problems are known to be *NP*-Hard.

Finally, the problem description is given and research assumptions are made.

CHAPTER FIVE
NEW STOCHASTIC LOCAL SEARCH ALGORITHMS

5.1 Introduction

Local Search (LS) techniques are algorithms that exploit the local neighborhood structures of a solution space in searching for the local optimal solution. These techniques start off with an initial random solution, and iteratively make local changes within the local neighborhood structures of the solution space in finding improved solutions. LS techniques try to determine the best neighbour surrounding the current solution in moving towards the local optima.

This research introduces three new Monte Carlo type LS metaheuristic algorithms. These algorithms have been developed by the author of this dissertation. The motivation for developing these new algorithms was to investigate search techniques that could be used to determine effective solutions to difficult optimization problems at low computational costs.

The three new LS metaheuristic algorithms introduced are called the Best Performance Algorithm (BPA), the Iterative Best Performance Algorithm (IBPA) and the Largest Absolute Difference Algorithm (LADA). Each of these algorithms employs techniques that maintain updated lists' of their best solutions found, during an iterative process. By performing LS and using the best solutions found, improved solutions may possibly be determined. If improved solutions are found, then the lists' of the algorithms will get updated accordingly. It is still too soon to categorize the types of optimization problems that these algorithms will be most suitably applied to. Further research will need to be done to determine this. BPA, IBPA and LADA are developed to determine solutions for both continuous and combinatorial optimization problems.

Meanwhile, the ability of these algorithms in determining solutions to the ACP problems was tested and results obtained are presented in chapters seven and eight. To determine the relative merits of the solutions found, comparisons of the solutions obtained with two other well-known LS metaheuristic algorithms were undergone. These popular algorithms are Tabu Search (TS) and Simulated Annealing (SA). Their solutions are also compared to the solutions of four population based metaheuristic algorithms presented in chapter six. The remaining part of this chapter gives descriptions of the BPA, IBPA and LADA metaheuristic algorithms. The TS and SA techniques are also explained.

5.2 Best Performance Algorithm

The Best Performance Algorithm is modeled in relation to the competitive nature of professional athletes. Professional athletes desire to push the boundaries of their best performances within competitive environments. This occurs for several reasons which could be personal and/or financial, amongst others. However, to give off their best performances, the athletes need to strategize and practice. Strategizing and practice will help athletes improve their talents by assisting them in the development of refined skills. These refined skills will enable athletes to perform at their best, within competitive environments, irrespective of their sporting disciplines.

An effective strategy used in improving performance is through the use of technology. Technology can be used to identify the weaknesses and strengths of athletes in them delivering a performance. By identifying and strengthening the weaknesses, or even developing new techniques in delivering a performance, an athlete could possibly register improved performances in being competitive. One way to identify the athletes' weaknesses and strengths is to maintain an archive or a collection of the athletes' best registered performances. This collection will provide a reference for athletes to review the previous best performances that they delivered. Once weaknesses are identified, appropriate changes can be made to the techniques used in delivering that performance. This will help the athlete develop refined skills, improving the chances of delivering better performances. Best performances can include those performed within competitive environments and even those during training sessions. The implementation of the BPA is modeled on the idea of an athlete maintaining a collection of a limited number of his/her best performances.

BPA is implemented by maintaining a sorted list of the individual athletes' best performances. This list is called the Performance List (PL). PL only maintains a limited number of the best registered performances, as the athlete may only be interested in working with a limited number of his/her best performances. Performances are arranged according to the quality of the performance delivered. The better the quality of a performance, the higher up on the list it is. The quality of a performance is a measure of the result obtained in executing that performance.

In trying to develop refined skills or possibly determine a new technique, which may possibly lead to an improved performance, the athlete will review a performance from the PL and will seek to make appropriate changes. By making slight changes (performing LS) to the way a reviewed performance was delivered, an improved technique may be determined which may lead to a better quality performance. If an improved technique is found, then the PL will be updated with this performance, provided that it at least improves on the worst performance on the PL. When an improved performance gets inserted into the PL, the worst performance is removed. The sorted order of the PL must always be maintained. Any improved technique that produces a performance which results in the quality of that performance being identical to that of another performance, which is already registered on the PL, will not get considered.

After making slight changes to the techniques used by the athlete in delivering his/her previous best performances, the athlete may want to continue making slight changes to those updated techniques, and so on and so forth for as long as he/she would like to. If improved techniques are found along the way which leads to improved performances, then the PL will get updated accordingly. If the athlete wants to work with another performance from the PL then the athlete will choose to do so. After a sufficient amount of strategizing and implementation the athlete will determine the best technique to use which will allow him to perform at his best.

From a heuristic perspective, the best performances recorded on the PL refer to the best solutions found by the metaheuristic algorithm. The performance/solution that the athlete will consider working with is called the “working” solution. Local changes are made to this working solution, in the hope of trying to determine an improved solution. If this *updated* working solution at least improves on the *worst* solution found on the PL then the PL will get updated. The athlete will continue working with this updated working solution or choose another solution from the PL to be its new working solution for the next iteration, given a certain probability. This probability symbolizes the athletes’ willingness to continue working with an updated working solution or not.

PL will always only get updated with solutions that provide unique performance results. This will prevent the algorithm from working with duplicate solutions that produce identical results. After a predetermined number of iterations are completed, the best solution found will be representative of the best technique determined by the athlete. This best solution will be the first solution registered on the PL. The algorithm for BPA is given in Algorithm 5.2.1 below. The flowchart diagram is given in Figure 5.2.1.

Algorithm 5.2.1: The Best Performance Algorithm

---

1. Set the index variable, $index = 0$
2. Set the size of the Performance List , $listSize$
3. Initialize probability, p_a
4. Populate the Performance List (PL) with random solutions
5. Calculate the fitness values of the solutions in PL, i.e. $PL\_Fitness$
6. Sort PL and $PL\_Fitness$ according to $PL\_Fitness$
7. Initialize $working$ to PL_{index}
8. **for** i to $noOfIterations$ **do**

8.1. $working$ = Perform_Local_Search($working$)

8.2. $f\_working$ = Evaluate ($working$)

8.3. **if** $f\_working$ better then $PL\_Fitness_{listSize-1}$ **then**

8.3.1. Update PL with $working$

8.3.2. Update $PL\_Fitness$ with $f\_working$

8.4. **end if**

8.5. **if** random[0,1] > p_a **then**

8.4.1. $index$ = Select index, e.g. Random[0,$listSize$]

8.4.2. $working = PL_{index}$

8.6. **end if**

9. **end for**

10. **return** PL_0

5.3 Iterative Best Performance Algorithm

With the BPA, an athlete determines improved techniques by making slight changes to the techniques used to deliver a limited number of the athletes' best performances (refer to section 5.2). At different iterations of the algorithm, the performance/solution chosen to be worked with will either be a new performance selected from the Performance List (PL) or the performance obtained from the previous iteration. Working with the performance from the previous iteration determines the willingness of the athlete to continue working with the previous performance. This willingness is represented by a predetermined probability variable in the algorithm. Given this probability, the algorithm either works with a previous performance or not.

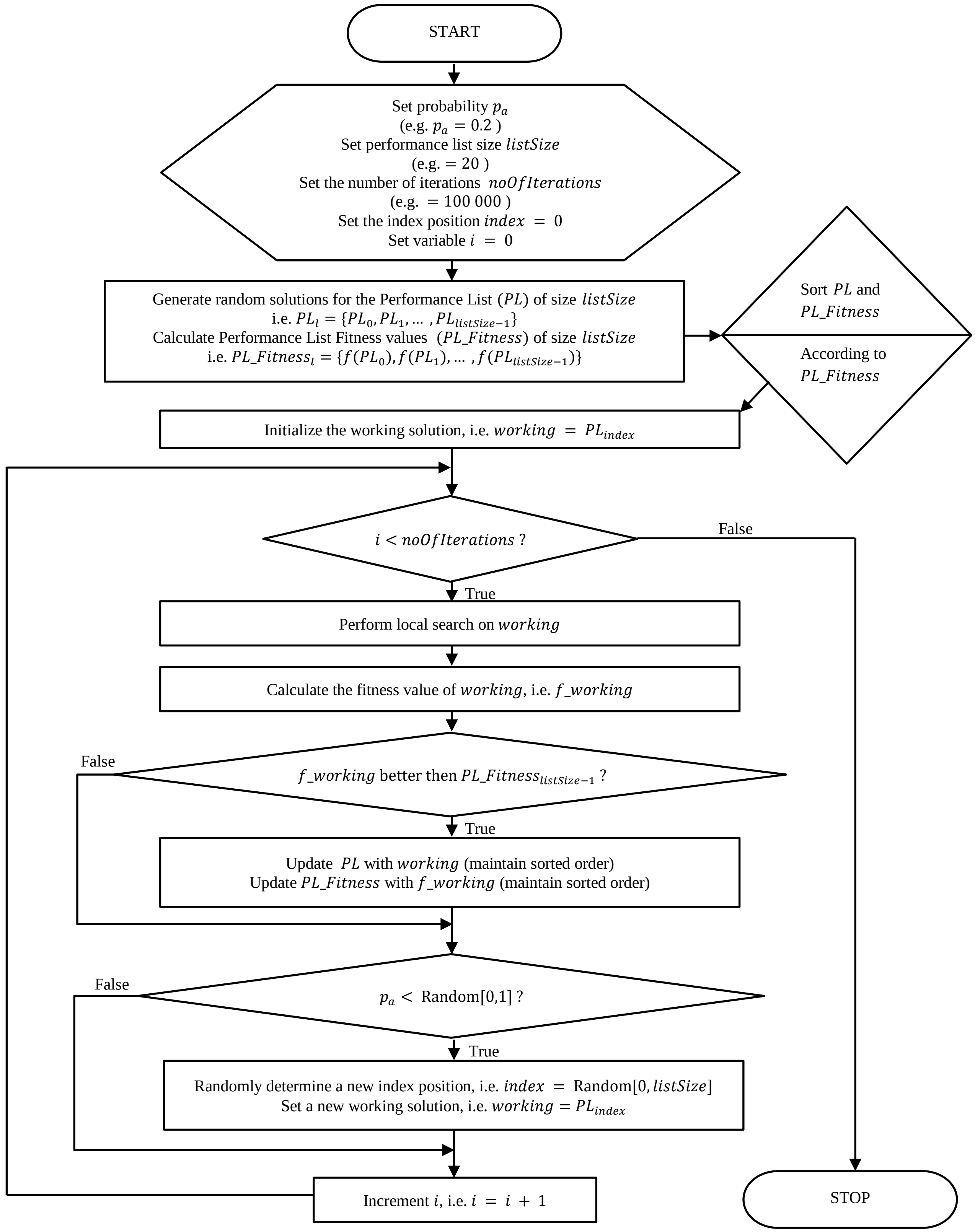

Figure 5.2.1: Flowchart of the Best Performance Algorithm

The Iterative Best Performance Algorithm (IBPA) is modeled on the same principles as the BPA. However, with the IBPA the athlete will continue working with the *same* performance for a specified amount of time. This performance is viewed as a reference performance. Using this reference performance, the athlete will make slight changes to the technique used to deliver that performance in the hope of trying to determine improved techniques. The athlete will continue doing this for a specified amount of time, in order to be satisfied that enough attempts were made in working with an individual performance. After the athlete is done working with a reference performance, another reference performance will be chosen from the PL. In working with these reference performances, improved techniques may be determined along the way. These improved techniques may lead to improved performances being delivered. If improved performances are delivered then the PL will be updated accordingly.

In implementing the IBPA, the reference performance is considered the "current" solution. This current solution remains the same for a predetermined number of iterations. This iteration count will be referred to as the 'steps per change'. The steps per change remain constant for the current solution worked with, for the number of current solutions that the athlete is willing to work with. The number of current solutions that the athlete is willing to work with is also specified by a predetermined number of iterations. This iteration count is referred to as the 'number of iterations'.

For each step per change, local search is performed on the current solution. This will generate a "working" solution. Similar to BPA, if the working solution is at least better than the worst solution on the PL, then the PL will get updated accordingly. After the number of steps per change is completed, in working with the current solution, another current solution will be chosen for the next set of steps per change. This process will continue until the number of iterations is complete. After the number of iterations is completed, the best solution determined will be the first solution on the PL. This solution is representative of the best technique determined by the athlete. The algorithm for IBPA is given in Algorithm 5.3.1 below. The flowchart diagram is given in Figure 5.3.1

Algorithm 5.3.1: The Iterative Best Performance Algorithm

1. Set the index variable, $index = 0$
2. Set the size of the Performance List , $listSize$
3. Populate the Performance List (PL) with random solutions
4. Calculate the fitness values of the solutions in PL, i.e. $PL\_Fitness$
5. Sort PL and $PL\_Fitness$ according to $PL\_Fitness$
6. Initialize $current$ to PL_{index}
7. **for** i to $noOfIterations$ **do**
   7.1. **for** j to $stepsPerChange$ **do**
      7.1.1. $working$ = Perform_Local_Search($current$)
      7.1.2. $f\_working$ = Evaluate($working$)
      7.1.3. **if** $f\_working$ better then $PL\_Fitness_{listSize-1}$ **then**
         7.1.3.1. Update PL with $working$
         7.1.3.2. Update $PL\_Fitness$ with $f\_working$
      7.1.4. **end if**
   7.2. **end for**
   7.3. $index$ = Select $index$, e.g. Random[0,$listSize$]
   7.4. $current = PL_{index}$ ($current_i \neq current_{i-1}$)
8. **end for**
9. **return** PL_0

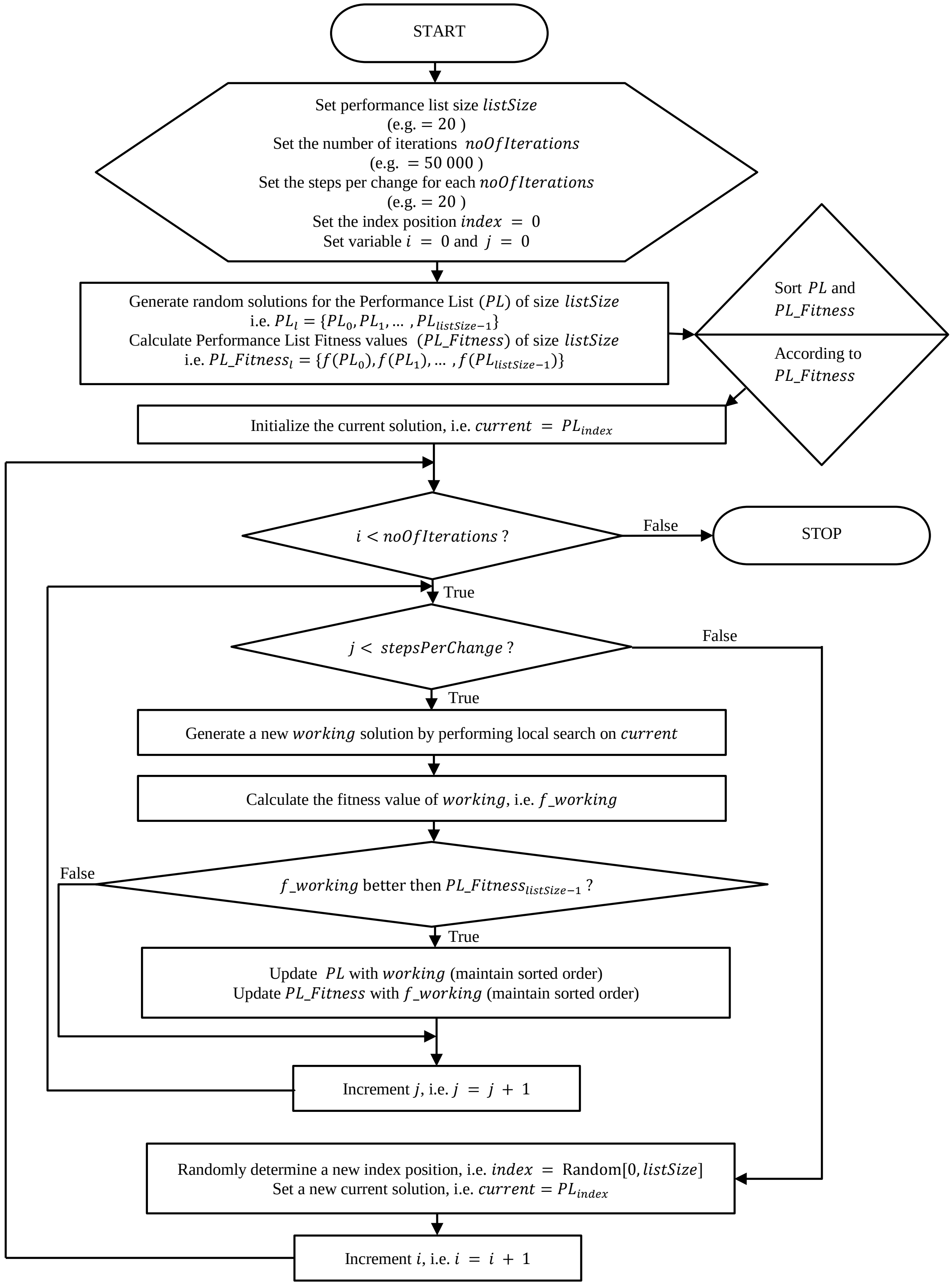

Figure 5.3.1: Flowchart of the Iterative Best Performance Algorithm

5.4 Largest Absolute Difference Algorithm

Difference, in mathematical terms, is the quantity which remains after one quantity is subtracted from another. An example is when the number ‘3’ is subtracted from the number ‘6’. The remainder is equivalent to -3. The remainder is negative because 3 is less than 6.

The *absolute difference* between two real numbered values x and y is the absolute value of their difference. It is denoted by $|x - y|$ and is mathematically defined as follows;

$$|x - y| = \begin{cases} (x - y), & iff\ (x - y) \geq 0 \\ -(x - y), & iff\ (x - y) < 0 \end{cases} \tag{5.1}$$

The absolute difference will always be positive or zero (if $x \equiv y$). On a real line it can be seen as the magnitude or difference between points x and y. This can be seen in Figure 5.4.1.

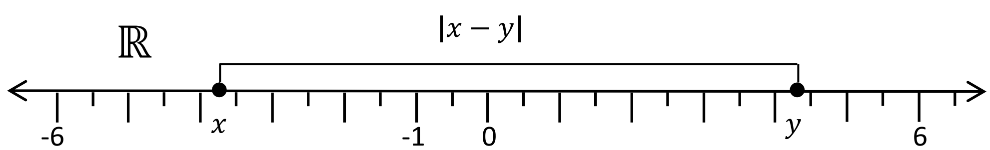

Figure 5.4.1: The absolute difference between the values x and y

The Largest Absolute Difference Algorithm (LADA) is modeled on the ability to calculate an absolute difference between real numbers.

During an optimization process, a solution vector $x \in \theta \subseteq \mathbb{R}^p$ (refer to chapter 2) is the input vector to the objective function f. x is the p-dimensional vector of design variables of f, i.e. $x = \{x_1, \dots, x_p\}$. Design variables can be continuous or discrete depending on the type of optimization problem. The values of the design variables will determine the state (or quality) of the objective function within the domain of the solution space. Several solutions can exist depending on the different values of the design variables. By taking two of these solutions x_i and x_j, a vector of absolute differences (d) can be determined by calculating the absolute differences of the values of the *adjacent elements* of vectors x_i and x_j. d is determined using equation 5.2 below.

$$d_k = |x_{i,k} - x_{j,k}| \quad for\ k = 1, \dots, p \tag{5.2}$$

The elements of d is indicative of how far away from each other the adjacent elements of the solution vectors x_i and x_j are. The indices of d, which are indicative of the smallest absolute differences, represent the indices of x_i and x_j that are most similar. The indices of d with the largest absolute differences represent the indices of x_i and x_j that are least similar. By performing local search on the adjacent elements of x_i and x_j, indexed by the largest absolute differences of d, new solution vectors x_i' and x_j' can be determined. If these new 'child' solutions improve on their 'parent' solutions then these solutions will be drawn closer together in moving towards the global optimum. By performing this local search technique on a population of solutions, the solutions will begin to converge to the global optimum in an iterative way.

LADA is implemented by maintaining a population of solutions in a list called the Solutions List (SL). SL must at least be greater than or equal to 2. Also, the best solution found in SL must be recorded in a variable called "best". LADA is executed for a specified number of iterations. At each iteration l, two solutions x_i and x_j will be randomly selected from SL ($i \neq j$). x_i and x_j gets copied respectively into their "working" variables $working_i$ and $working_j$. Using $working_i$ and $working_j$ the vector of absolute differences d_l can be determined. To implement local search, using d_l, the number of largest absolute differences to be worked with must be specified. This is given by the variable m, where $0 < m \leq n$. Having determined d_l, and knowing m, two new child solutions are generated by making permissible changes to $working_i$ and $working_j$. If $working_i$ provides a better quality solution than SL_i, then SL_i will be replaced by $working_i$. Similarly, if $working_j$ improves on SL_j, SL_j will be replaced by $working_j$. If $working_i$ or $working_j$ improves on $best$ then $best$ must be updated accordingly. The quality of the solutions of $working_i$ and $working_j$ must not be identical to the quality of any other solution found in SL. Disallowing identical quality solutions ensures the uniqueness of the solutions listed in the SL. After the specified number of iterations is completed, the best solution found will be recorded in best. The algorithm for LADA is given in Algorithm 5.4.1 below. The flowchart is given in Figure 5.4.2.

Algorithm 5.4.1: The Largest Absolute Difference Algorithm

1. Set the size of the Solutions List, $listSize$
2. Populate the Solutions List (SL) with random solutions
3. Calculate the fitness values of the solutions in SL, i.e. $SL\_Fitness$
4. Set the no. of absolute differences to consider, m
5. Set the best solution ($best$) and best fitness (f_best) using $SL\_Fitness$
6. **for** i to $noOfIterations$ **do**
    6.1. $index1$ = Select $index1$, e.g. Random[0,$listSize$]
    6.2. $index2$ = Select $index2$, e.g. Random[0,$listSize$] ($index1 \neq index2$)
    6.3. $working\_1 = SL_{index1}$
    6.4. $working\_2 = SL_{index2}$
    6.5. $d = |working\_1 - working\_2|$
    6.6. Perform_LS($working\_1, working\_2, d, m$)
    6.7. $f\_working\_1$ = Evaluate($working\_1$)
    6.8. $f\_working\_2$ = Evaluate($working\_2$)
    6.9. **if** $f\_working\_1$ better then $SL\_Fitness_{index1}$ **then**
        6.9.1. $SL_{index1} = working\_1$
        6.9.2. $SL\_Fitness_{index1} = f\_working\_1$
        6.9.3. **if** $f\_working\_1$ better then f_best **then**
            6.9.3.1. $best = working\_1$
            6.9.3.2. $f\_best = f\_working\_1$
        6.9.4. **end if**
    6.10. **end if**
    6.11. **if** $f\_working\_2$ better then $SL\_Fitness_{index2}$ **then**
        6.11.1. $SL_{index2} = working\_2$
        6.11.2. $SL\_Fitness_{index2} = f\_working\_2$
        6.11.3. **if** $f\_working\_2$ better then f_best **then**
            6.11.3.1. $best = working\_2$
            6.11.3.2. $f\_best = f\_working\_2$
        6.11.4. **end if**
    6.12. **end if**
7. **end for**
8. **return** $best$

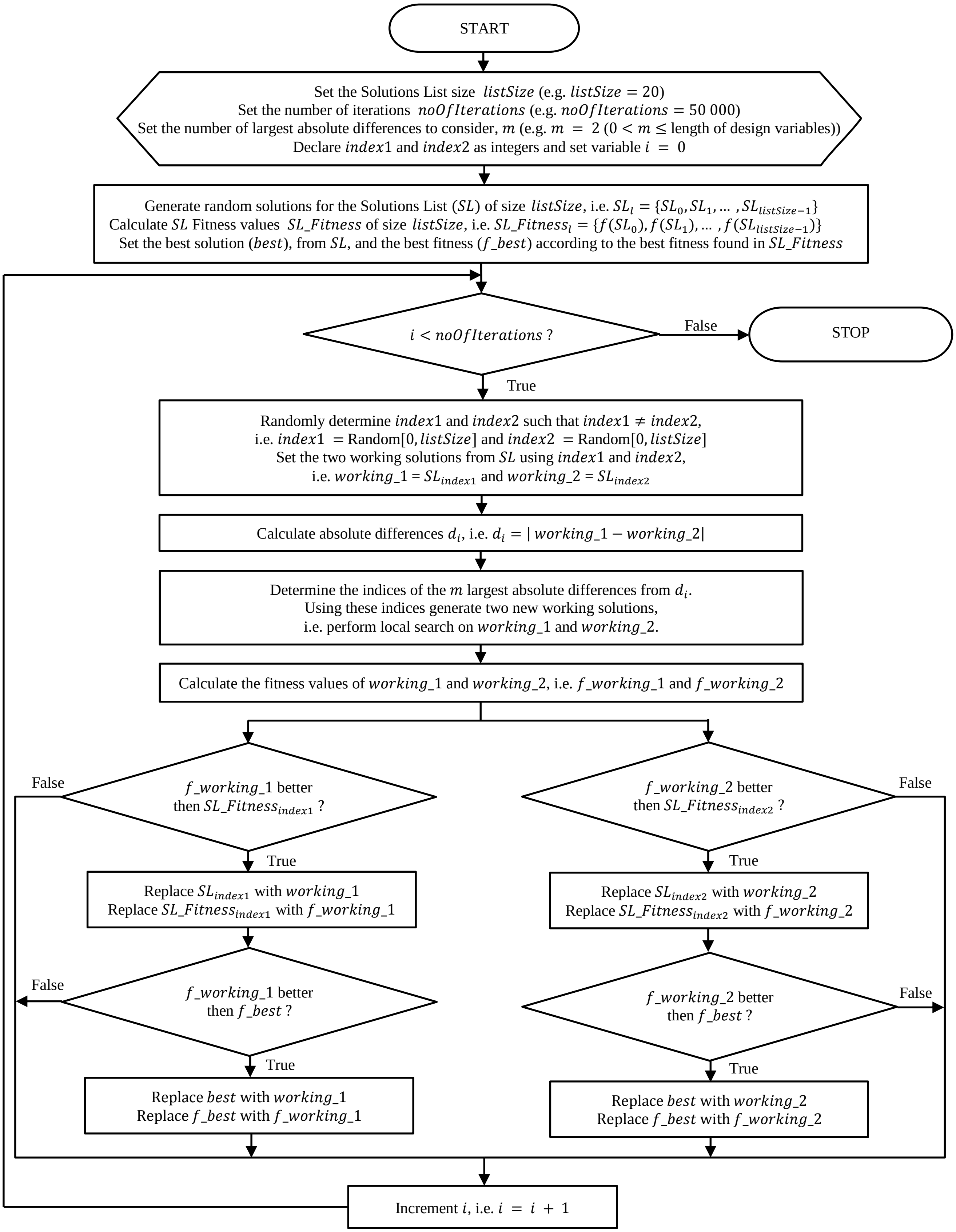

Figure 5.4.2: Flowchart of the Largest Absolute Difference Algorithm

5.5 Tabu Search

TS is based on the idea of something that should not be interfered with (Glover, 1989; Glover, 1990). TS implements this idea by recording a specific number of unique best solutions found in a list called the Tabu List (TL). If a new solution is found that improves on the solutions recorded in the TL, the new solution gets added to the TL. Any new solutions found that is identical to those already registered in the TL will not be considered. This eliminates the possibility of exploiting identical moves.

TS also maintains a record of the "best" overall solution. Using a "current" solution, TS generates a list of candidate solutions, which are local to the current solution. The new candidate solutions determined must be cross referenced against the TL. This will eliminate the possibility of repeating identical moves. Once the candidate list is determined, the best candidate solution from the list can be found. This best candidate solution becomes the new current solution for the next iteration. If this new current solution improves on the best solution found so far, then it also gets recorded as the best solution and gets inserted into the TL. The TL is usually updated using the First In First Out technique.

Generating new solutions is done in a deterministic way, using local search. This process continues iteratively for a specific number of iterations. The algorithm for TS is given in Algorithm 5.5.1 below.

Algorithm 5.5.1: Tabu Search

---

1. Generate an initial random solution $= best$
2. Set $current = best$
3. Evaluate the fitness of $best = f\_best$
4. Set the fitness of $current$ $(f\_current) = f\_best$
5. Set the size of the Tabu List, $tabuListSize$
6. Set the size of the Candidate List, $candidateListSize$
7. Initiate the Tabu List TL and the $CandidateList$

8. **for** i to $noOfIterations$ **do**
    8.1. $CandidateList$ = Generate_List($current$)
    8.2. $current$ = Find_Best_Candidate($CandidateList$)
    8.3. $f\_current$ = Evaluate($current$)
    8.4. **if** $f\_current$ better then f_best **then**
        8.4.1. $f\_best = f\_current$
        8.4.2. $best = current$
        8.4.3. Update TL with $current$
    8.5. **end if**
9. **end for**
10. **return** $best$

5.6 Simulated Annealing

SA (Kirkpatrick, 1983; Tan, 2008) models the annealing process when heated metal begins to cool. The hotter metal gets when heated, the more volatile its atomic structure will become. This will result in a weakened and unstable structure. However, when the heated metal begins to cool, the highly energized metallic atoms lose energy and the structure begins to stabilize. When the metal is completely cooled, an equilibrium state is reached. The cooling process must be slow for the annealing to be successful. Reaching an equilibrium state is symbolic of an optimal solution being found for optimization problems.

SA starts off with randomly generated, but equivalent, “best”, “current” and “working” solutions. It starts off with an initial temperature T and then decreases by a constant factor α, until it reaches its final temperature F. At each reduced temperature $T \times \alpha$, SA iteratively searches for local solutions to the current solution. This constitutes the working solution. If the working solution is better than the current solution, the current solution is replaced by this working solution. If this current solution is better than the best solution, then the best solution becomes the current solution. The worst working solutions can replace the current solution, given a certain probability. This strategy reduces the chances of premature convergence. This process continues until F is reached. F symbolizes an equilibrium state being reached where the best solution found will be given. The algorithm for SA is given in Algorithm 5.6.1 below.

Algorithm 5.6.1: Simulated Annealing

1. Generate an initial random solution = $best$
2. Set $current = working = best$
3. Evaluate the fitness of $best = f\_best$
4. Set the fitness of $current$ ($f\_current$) and the fitness of $working$ ($f\_working$) $= f\_best$
5. Initiate starting temperature T and final temperature F
6. **while** $T \geq F$ **do**
    6.1. **for** i to $stepsPerChange$ **do**
        6.1.1. $working$ = Generate_Solution($current$)
        6.1.2. $f\_working$ = Evaluate($working$)
        6.1.3. **if** $f\_working$ better then $f\_current$ **then**
            6.1.3.1. $use\_solution$ = true
        6.1.4. **else**
            6.1.4.1. Calculate acceptance probability P
            6.1.4.2. **if** $P >$ random[0,1] **then**
                6.1.4.2.1. $use\_solution$ = true
            6.1.4.3. **end if**
        6.1.5. **end else**
        6.1.6. **if** $use\_solution$ **then**
            6.1.6.1. $use\_solution$ = false
            6.1.6.2. $f\_current$ = $f\_working$
            6.1.6.3. $current$ = $working$
            6.1.6.4. **if** $f\_current$ better then f_best **then**
                6.1.6.4.1. $best$ = $current$
                6.1.6.4.2. f_best = $f\_current$
            6.1.6.5. **end if**
        6.1.7. **end if**
    6.2. **end for**
    6.3. Update T according to cooling schedule
7. **end while**
8. **return** $best$

5.7 Conclusion

This chapter introduces three new Monte Carlo type Local Search (LS) metaheuristic algorithms. These algorithms are the Best Performance Algorithm (BPA), the Iterative Best Performance Algorithm (IBPA) and the Largest Absolute Difference Algorithm (LADA). BPA and IBPA are modeled on the competitive nature of professional athletes in trying to improve on their best registered performances. LADA is modeled on the ability to calculate the absolute difference between two real numbers.

The techniques used by these algorithms maintain updated lists' of their best solutions found. By maintaining collections of these best solutions, the algorithms are directed towards determining more improved solutions in performing LS.

BPA, IBPA and LADA will be used to determine solutions to the ACP problems presented in chapters seven and eight. To determine the relative merits of the solutions found, solutions will be compared to the solutions of two well-known LS metaheuristic algorithms. These algorithms are Tabu Search (TS) and Simulated Annealing (SA). Both TS and SA have been briefly described in this chapter.

CHAPTER SIX
POPULATION BASED TECHNIQUES FOR THE ANNUAL CROP PLANNING PROBLEM

6.1 Introduction

Swarm Intelligence (SI) is research that is inspired by observing the naturally intelligent behaviour of swarms of biological agents, within their environments. The swarms are typically made up of simple agents that perform simple tasks while interacting with each other and their environment. However, without any central control structure directing their movements they seem to interact intelligently, and in an independent way, in achieving their overall objectives (Blum and Merkle, 2008). These observations have led to the development of many effective SI optimization algorithms. These algorithms typically represent the individual behavior of the biological agents which are represented by a set of simple rules. Examples of swarm systems studied in literature include colonies of ants, wasps, termites and bees, flocks of birds, school of fish and herds of animals, amongst others (Blum and Merkle, 2008).

SI algorithms have been effective in providing solutions to many NP type optimization problems in literature. This research investigates the effectiveness of employing three relatively new SI metaheuristic algorithms, in determining solutions to the ACP problems presented in chapters seven and eight. The algorithms investigated are Cuckoo Search (CS), Firefly Algorithm (FA) and Glowworm Swarm Optimization (GSO). To determine the relative merits of their solutions found, their solutions will be compared to that of a well-known population based metaheuristic algorithm. This algorithm is the Genetic Algorithm (GA).

GA is a global search metaheuristic algorithm. Global search techniques attempt to determine the single best local optimum solution from the set of local optima that exist within the solution space. The single best local optimum solution is the global optimum solution.

Practically, there are no global search algorithms that exist that guarantee determining optimal solutions for all NP type optimization problems. The aim of global search algorithms is therefore to determine the best local optimum solution that can be found within P.

The subsections below describe CS, FA, GSO and the GA.

6.2 Cuckoo Search

CS (Yang, 2010) is inspired by the parasitism of some cuckoo bird species. These birds aggressively reproduce and then abandon their eggs in the nests of other host bird species. Some host bird's behave aggressively and throw away the alien eggs after discovering an intrusion. Others simply leave their nests and build new nests elsewhere.

Each egg in the host bird's nest represents a possible solution. The goal of the CS algorithm is to replace a not-so-good solution in the host bird's nest with a potentially better solution. This is represented by a newly-laid egg. There are three guiding rules governing the CS algorithm. These include:

1. Each bird lays one egg at a time. The egg gets placed randomly amongst the host bird's nests.
2. The nest with the highest fitness value will get carried over to the next generation.
3. The number of host bird nests is fixed. The probability of a host bird discovering an intrusion is set at a constant value of $p_a \in [0,1]$.

In generating a new solution, the random-walk is best performed in using levy flights. The levy flight of cuckoo i is performed using equation (6.1).

$$x_i(t+1) = x_i(t) + s\delta \tag{6.1}$$

Here, δ is drawn from a standard normal distribution with mean 0 and standard deviation of 1. δ determines the direction of movement. s is the step size. This determines the distance of the random walk. Determining s is tricky. If s is too big then $x_i(t+1)$ will be too far away from $x_i(t)$. If s is too small then $x_i(t+1)$ will be too close to $x_i(t)$ to be significant enough. One of the most efficient algorithms used to calculate s is Mantegna's algorithm (Yang, 2010). Using Mantegna's algorithm, s can be calculated by using equation (6.2).

$$s = \frac{u}{|v|^{1/\beta}} \tag{6.2}$$

Here, u and v are drawn from a normal distribution, and $0 < \beta \leq 2$.

The algorithm for Cuckoo Search is given in Algorithm 6.2.1 below.

Algorithm 6.2.1: Cuckoo Search

---

1. Generate an initial random solution of n host bird nests = $nest$ (for $i = 1, \dots, n$)

2. Evaluate the fitness of $nest_i$ by summing the values of the solutions of each egg in $nest_i$, i.e. $f(nest_i)$.

3. Find the best fitness ($bestFitness$) and best nest ($bestNest$) from $nest$

4. $bestFitnessOverall$ = $bestFitness$

5. $bestNestOverall$ = $bestNest$

6. **while** $t < noOfIterations$ **do**

 6.1. Generate $newNest$, using $nest$ and $bestNest$ in performing levy flights

 6.2. Get $bestNest$ by performing these steps

  6.2.1. **if** $f(newNest_i) > f(nest_i)$ **then**

   6.2.1.1. $f(nest_i) = f(newNest_i)$

   6.2.1.2. $nest_i$= $newNest_i$

  6.2.2. **end if**

  6.2.3 Evaluate $f(nest_i)$ to determine $bestFitness$ and $bestNest$

 6.3. $t = t + n$

 6.4. Generate $newNest$, using $nest$ and p_a. Here, a fraction of the worst solutions are replaced with new solutions for each $nest_i$

 6.5. Determine $bestNest$ again using step 6.2.

 6.6. $t = t + n$

 6.7. **if** $bestFitness$ > $bestFitnessOverall$ **then**

6.7.1. $bestFitnessOverall = bestFitness$

6.7.2. $bestNestOverall = bestNest$

6.8. **end if**

7. **end while**

8. **return** $bestNestOverall$

6.3 Firefly Algorithm

FA (Yang, 2010) is inspired by the ability of fireflies to emit light (bioluminescence) in order to attract other fireflies for mating purposes. There are three guiding rules governing this algorithm. These include:

1. Fireflies are attracted towards brighter fireflies, regardless of their sex.
2. The attractiveness of a firefly is related to its brightness. However, it is assumed that this brightness decreases with distance. The brightest firefly moves randomly.
3. The brightness of the firefly is a function of the problems' objective.

Attractiveness: The attractiveness of a firefly is given by equation (6.3).

$$\beta(r) = \beta_0 exp^{-\gamma r^2} \tag{6.3}$$

Here, r is the distance between any two fireflies. β_0 represents the initial attractiveness at $r = 0$. γ is an absorption coefficient. It controls the decrease in the intensity of light.

Movement: The movement of a less attractive firefly i towards a more attractive firefly j is given by equation (6.4).

$$x_i = x_i + \beta_0 exp^{-\gamma r_{ij}^2}\left(x_j - x_i\right) + \alpha\left(rand - \frac{1}{2}\right) \tag{6.4}$$

Here, x_i is the current position of the firefly within the solution space. The combination of the elements in the second term represents the firefly's attractiveness, as seen by the other fireflies. The third term represents a random adjustment in the movement of the firefly. α is a scaling parameter, $\alpha \in [0,1]$. $rand$ is a uniformly distributed random number, $rand \in (0,1)$.

r_{ij} represents the distance between fireflies i and j. It is calculated using the Cartesian distance (Yang, 2010) given in equation (6.5).

$$r_{ij} = \sqrt{\sum_{k=1}^{d}(x_{ik} - x_{jk})^2} \tag{6.5}$$

The algorithm for FA is given in Algorithm 6.3.1 below.

Algorithm 6.3.1: Firefly Algorithm

1. Initialize α, β_0, γ and $noOfIterations$

2. Initialize n fireflies = $fireflyLocations$(for $i = 1 \dots n$)

3. The light intensity of $fireflyLocations_i = fireflyFitness_i$

4. **for** l till $noOfIterations$ **do**

   4.1. **for** i till n **do**

      4.1.1 $fireflyFitness_i$ = Evaluate($fireflyLocations_i$)

   4.2. **end for**

   4.3. Sort $fireflyLocations$ and $fireflyFitness$ according to $fireflyFitness$

   4.4. $bestFireflyFitness$ = $fireflyFitness_0$

   4.5. $bestFireflyLocation$ = $fireflyLocations_0$

   4.6. Move fireflies to new locations by performing these steps

      4.6.1. **for** i till n **do**

         4.6.1.1. **for** j till n **do**

            4.6.1.1.1. **if** $fireflyFitness_i < fireflyFitness_j$ **then**

               4.6.1.1.1.1. Calculate r_{ij}

               4.6.1.1.1.2. Calculate $\beta(r)$

               4.6.1.1.1.3. Update $fireflyLocations_i$

            4.6.1.1.2. **end if**

         4.6.1.2. **end for**

      4.6.2. **end for**

5. **end for**

6. **return** $bestFireflyLocation$

6.4 Glowworm Swarm Optimization

GSO (Krishnand and Ghose, 2009a; Krishnand and Ghose, 2009b) is inspired by the natural behaviour of glow-worms in emitting a luminescent property, called luciferin, in order to attract other glow-worms. Glow-worms with larger emissions of luciferin are considered more attractive. Glow-worms move towards a brighter glow-worm, if it lies within its range of view.

Initially, glow-worms are distributed randomly throughout the solution space. At any point in time t, the state of a glow-worm i is represented by its luciferin level $l_i(t)$, its position $x_i(t)$ and its vision range $r_i(t)$. During each iteration, these variables are updated and it describes the movement of the glow-worms within the solution space.

The luciferin update is given by equation (6.6)

$$l_i(t+1) = (1-\rho)l_i(t) + \gamma\mathcal{J}\big(x_i(t)\big) \tag{6.6}$$

Here, ρ is the luciferin decay constant ($0 < p < 1$). γ is the luciferin enhancement constant. $\mathcal{J}\big(x_i(t)\big)$ is the evaluation of the objective function, at time t.

To update the position of each glow-worm i, a set of neighbours $N_i(t)$ need to be determined. A glow-worm j is considered a neighbour of glow-worm i, if j falls within i's vision range $r_i(t)$, and if $l_i(t) < l_j(t)$. A glow-worm j is then selected from $N_i(t)$, using roulette wheel selection. Glow-worm i then moves in the direction of glow-worm j using equation (6.7).

$$x_i(t+1) = x_i(t) + st * \left\{\frac{x_j(t)-x_i(t)}{\|x_j(t)-x_i(t)\|}\right\} \tag{6.7}$$

Here, st is a constant step size.

Lastly, the vision range $r_i(t)$ needs to be updated. It is updated using equation (6.8)

$$r_i(t+1) = min\{r_s, max[0, r_i(t) + \beta(N_d - |N_i(t)|)]\} \qquad (6.8)$$

Here, r_s, β and N_d are constant values. r_s is the maximum vision range and N_d is the maximum number of neighbour's that glow-worm i is allowed to have.

The algorithm for GSO is given in Algorithm 6.4.1 below.

Algorithm 6.4.1: Glowworm Swarm Optimization

1. Generate a population of n glow-worms = $glowworm$ (for $i = 1, \dots, n$)
2. Initialize the best fitness overall = $bestFitness$
3. Initialize the best location overall = $bestLocation$
4. **while** t till $noOfIterations$ **do**
   4.1. **for** i till n **do**
      4.1.1. Update luciferin of $glowworm_i$
   4.2. **end for**
   4.3. **for** i till n **do**
      4.3.1. Find $N_i(t)$
      4.3.2. **for** each $glowworm_j \in N_i(t)$ **do**
         4.3.2.1. Find probability: $p_{ij}(t) = \frac{l_j(t) - l_i(t)}{\sum_{k \in N_i(t)} l_k(t) - l_i(t)}$
      4.3.3. **end for**
      4.3.4. Select $glowworm_j$ using roulette wheel selection with $p_{ij}(t)$
      4.3.5. Update $glowworm_i$ location
      4.3.6. Update vision range
   4.4. **end for**
   4.5. **for** i till n **do**

4.5.1. **if** $glowworm_i.fitness > bestFitness$ **then**

4.5.1.1. $bestFitness = glowworm_i.fitness$

4.5.1.2. $bestLocation = glowworm_i.Location$

4.5.2. **end if**

4.6. **end for**

4.7. $t = t + 1$

5. **end while**

7. **return** $bestLocation$

6.5 Genetic Algorithm

GA (Holland, 1975) is inspired by the process of natural evolution. By modeling evolutionary processes such as selection, crossover and mutation a population of chromosomes (genotypes of the phenotypes or individuals) evolve from one generation to the next. Chromosomes are binary encoded for discrete optimization problems or real-value encoded for continuous optimization problems (Eiben and Smith, 2003).

GA starts off with an initial, randomly generated, population of chromosomes/solutions. Each solution has an associated fitness value which is indicative of the individuals' strength. Using these fitness values, pairs of solutions get stochastically selected from the current population, at each generation. However, if the fitness values are not used in selecting the pairs then the pairs of solutions get selected using randomization techniques. Using techniques such as crossover and mutation, these pairs of solutions will produce offspring solutions. The offspring solutions form the new population, which represent the next generation. This process will continue for a specified number of generations or until a satisfactory fitness value is found.

Selection is done using techniques such as the roulette wheel selection and random selection, amongst others (Eiben and Smith, 2003). Roulette wheel selection considers the fitness value of the solutions, while random selection does not. When pair of solutions gets selected, the crossover process generates offspring solutions which are a recombination of their parent solutions. Recombination is done using techniques such as n-point crossover, uniform crossover and arithmetic crossover, amongst others (Eiben and Smith, 2003). The genes of the offspring's get mutated given a certain probability. Mutation reduces the risk of premature convergence. Premature convergence occurs when the heuristic algorithm gets stuck within a local neighborhood structure of the solution space, in which case, the local optimal solution is not close enough to the global optimal solution.

The implementation of GA in this research was done using real-value encoding and uniform crossover. The algorithm of GA used in this research is given in Algorithm 6.5.1 below.

Algorithm 6.5.1: Genetic Algorithm

1. Generate an initial random population of n individuals = $population$ (for $i = 1, \dots, n$)
2. Initialize another population of size n, i.e. $newPopulation$
3. Evaluate the fitness of each individual $population_i$, i.e. $population.fitness_i$
4. Determine the best individual from $population$ using $population.fitness_i$ = $bestIndividual$
5. Set crossover rate = $cRate$
6. Set mutation rate = $mRate$
7. **for** i till $maxNoOfGenerations$ **do**
    7.1. $count = 0$
    7.2. **while** $count < n$ **do**
        7.2.1. Select parents
        7.2.2. Perform crossover using $cRate$
        7.2.3. Perform mutation using $mRate$
        7.2.4. Add offspring's to $newPopulation$

7.2.5. $count\ = count +\ 2$

7.3. **end while**

7.4. $population$ = $newPopulation$

7.5. $bestIndiv$ = find_Best_Individual($population$)

7.6. **if** $bestIndiv.fitness$ better than $bestIndividual.fitness$ **then**

7.6.1. $bestIndividual$ = $bestIndiv$

7.7. **end if**

8. **end for**

9. **return** $bestIndividual$

6.6 Conclusion

SI techniques are nature-inspired methods derived by observing the naturally intelligent behavior of swarms of biological agents, their interactions with each other and the environment in which they perform their tasks. These observations have led to the development of several metaheuristic algorithms. These algorithms have been successfully applied in determining solutions to several NP type optimization problems.

This chapter describes three relatively new SI metaheuristic algorithms. These algorithms are the Cuckoo Search (CS), the Firefly Algorithm (FA) and Glowworm Swarm Optimization (GSO). Similar to the LS metaheuristic algorithms presented in chapter five, these algorithms will also be investigated in determining solutions to the ACP problems presented in chapters seven and eight. To determine the relative merits of the solutions found by these algorithms, their solutions will be compared against the solutions of a well-known population based metaheuristic algorithm. This algorithm is the Genetic Algorithm (GA). GA has been briefly described in this chapter.

CHAPTER SEVEN
PERFORMANCE ON AN EXISTING IRRIGATION SCHEME

7.1 Introduction

The increased costs associated with agricultural production have resulted in an increase in the cost of food. These increased food prices have made food less affordable for people and has contributed to several types of social problems. These include poverty, malnutrition and disease, amongst others. The increased production costs have also made it more expensive for crop producers to produce crops. Given the limited resources available for crop production, and the increased production costs, it has become very important that optimized solutions be found to the problem of crop planning. Crop producers now require more returns per area of crops cultivated.

The costs associated with crop production (refer to chapter three) include labour costs, the cost of irrigated water, fertilizers, pesticides, equipment costs, transportation costs and storage costs, amongst others. The limited resources include land, irrigated water, financial limitations and other types of resources associated with crop production. In crop planning, optimized solutions need to be found in the allocation of the limited resources amongst the various competing crops that are to be grown. Due to the concern of excessive volumes of irrigated water wastage by the agricultural sector, solutions need to be found in making efficient irrigated water allocation decisions.

Determining optimized solutions to crop planning, at the level of an irrigation scheme, is referred to as Annual Crop Planning (ACP) in this research. The objective of determining solutions to this problem is to maximize the total gross profits earned at an irrigation scheme in producing all the crops required to be grown within a production year, in efficiently allocating resources.

This chapter introduces the problem of ACP for an *existing* irrigation scheme. To determine feasible solutions, the economic demand of the crops, the plant requirements, the climatic conditions, the available area of agricultural land and the various costs associated with agricultural production need to be considered. The solutions found must seek to optimize the resource allocations amongst the various competing crops being grown, in maximizing total gross profits while satisfying the conditions associated with the problem.

To determine solutions for the ACP problem at *existing* irrigation schemes, a new mathematical model is formulated and presented in this chapter. The objective is to maximize the total gross profits earned in efficiently allocating limited resources amongst the various competing crops being produced within a production year.

To determine solutions for this NP-Hard optimization problem, this chapter investigates the abilities of three new Local Search (LS) and three relatively new Swarm Intelligence (SI) metaheuristic algorithms in determining solutions. As presented in chapter five, the LS metaheuristic algorithms are the Best Performance Algorithm (BPA), the Iterative Best Performance Algorithm (IBPA) and the Largest Absolute Difference Algorithm (LADA). The SI algorithms (see chapter six) include Cuckoo Search (CS), the Firefly Algorithm (FA) and Glowworm Swarm Optimization (GSO). To determine the relative merits of the solutions found by these algorithms, their solutions will be compared with the solutions of Tabu Search (TS), Simulated Annealing (SA) and the Genetic Algorithm (GA). The solutions determined and comparisons made will indicate the possible strengths and/or weaknesses of the three new LS and three relatively new SI algorithms, in determining solutions. The solutions found will be valuable in being compared to the statistics of the current agricultural practices at the irrigation scheme.

7.2 ACP Model for an Existing Irrigation Scheme

The ACP model in this study is formulated as part of the objectives of this research work. The model is designed to maximize the total gross profits that can be earned from a given area of land, which has been allocated for crop production. The functions' objective makes efficient use of the limited resources available in determining the seasonal hectare allocations amongst the various competing crops that are to be grown within a production year. Feasible solutions must satisfy the multiple land and irrigated water allocation constraints that are associated with the objective function. To determine optimized solutions to irrigated water supply, precipitation must be considered.

In crop production, the crops cultivated are those that are grown throughout the year. These are the perennial crops, such as the tree bearing crops. Other crop types include the seasonal crops such as the summer, autumn and winter crops. Single-crop plots are allocated to those crops that are grown throughout the year. Double-crop plots are allocated to two different types of crops that are grown in sequence, within the year. Triple-crop plots are allocated to three different types of crops that are grown in sequence within a year, and so on.

The soil and climatic conditions are also important factors in crop planning. The selection of the crops to be produced should adapt well to the given environmental conditions of the area. This is important in determining optimal yields.

Application of irrigated water is also important. Application of too much or too little water will lead to sub-optimal plant growth. This will affect the yield of the crop. The soil is also sensitive to leaching due to excessive water application (Blaylock, 2004). Therefore, the seasonal irrigated water allocation amongst the various crops should be well planned.

The ACP mathematical model for determining solutions to the ACP problem at existing irrigation schemes is formulated as follows:

7.2.1. Indices

- k – Plot types. (1 = single-crop plots, 2 = double-crop plots, 3 = triple-crop plots, and so on).
- i – Indicative of the groups of crops that are grown in sequence throughout the year, on plot type k ($i = 1$ represents the 1st group of sequential crops, $i = 2$ represents the 2nd group of sequential crops, $i = 3$ represents the 3rd group of sequential crops, and so on).
- j – Indicative of the individual crops grown at stage i, on plot k.

7.2.2. Input Parameters

- l – Number of different plot types.
- N_k – Number of groups of sequential crops grown within a year, on plot k.
- M_{ki} – Number of different crops grown at stage i, on plot k.
- L_{ki} – Total area of land allocated for crop production at stage i, on plot k.
- F_{kij} – Average fraction per hectare of crop j, at stage i, on plot k, which needs to be irrigated (1 = 100% coverage, 0 = 0% coverage).
- R_{kij} – Averaged rainfall estimates that fall during the growing months for crop j, at stage i, on plot k.
- CWR_{kij} – Crop Water Requirements of crop j, at stage i, on plot k.
- T – Total hectares of land allocated for crop production.
- A – Volume of irrigated water that can be supplied per hectare (ha^{-1}).
- P – Price of irrigated water m^{-3}.
- O_{kij} – Other operational costs ha^{-1} of crop j, at stage i, on plot k. This cost excludes the cost of irrigation.
- YR_{kij} – The amount of yield expected in tons per hectare (t ha^{-1}) of crop j, at stage i, on plot k.
- MP_{kij} – Expected producer prices of crop j, at stage i, on plot k.

- Lb_{kij} – Lower bound for crop j, at stage i, on plot k.
- Ub_{kij} – Upper bound for crop j, at stage i, on plot k.

7.2.3. Calculated Parameters

- TA – Total volume of irrigated water that can be supplied to the given area of land, within a year ($TA = T * A$).
- IR_{kij} – Volume of irrigated water estimates that should be applied to crop j, at stage i, on plot k. ($IR_{kij}m^3 = (CWR_{kij}m - R_{kij}m) * 10000m^2 * F_{kij}$).
- C_IR_{kij} – The cost of irrigated water ha^{-1} of crop j, at stage i, on plot k. ($C\_IR_{kij} = IR_{kij} * P$).
- C_{kij} – Variable costs ha^{-1} of crop j, at stage i, on plot k. ($C_{kij} = O_{kij} + C\_IR_{kiij}$).
- B_{kij} – Gross margin that can be earned ha^{-1} for crop j, at stage i, on plot k. ($B_{kij} = MP_{kij} * YR_{kij} - C_{kij}$).

7.2.4. Variables

- X_{kij} – Area of land, in hectares, that can be feasibly allocated to crop j, at stage i, on plot k.

7.2.5. Objective Function

Maximize

$$f = \sum_{k=1}^{l} \sum_{i=1}^{N_k} \sum_{j=1}^{M_{ki}} X_{kij} B_{kij} \tag{7.1}$$

In Equation 7.1, k represents the plot types. $k = 1$ indicates the single-crop plots, $k = 2$ indicates the double-crop plots, and so on. For each plot type k, i is indicative of the number of groups of crops that are grown in sequence throughout the year. For $k = 1$, N_k (or N_1) will be equivalent to 1. This will represent the group of crops that are grown all year round. For $k = 2$, $N_k = 2$. This will represent two groups of crops that are grown in sequence throughout the year. These are the summer and winter crops. The explanation is similar for $k = 3$, and so on. For each sequential crop group i, grown on plot k, j will represent the individual crops grown.

For $k = 1$ and $i = 1$, j will be indicative of all the perennial crops grown. For $k = 2$ and $i = 1$, j will be indicative of all the summer crops grown. For $k = 2$ and $i = 2$, j will be indicative of all the winter crops grown, and so on.

Equation 7.1 is subject to the land and irrigated water allocation constraints given in sections 7.2.6 and 7.2.7 below. The gross benefits B_{kij} that can be earned per crop must also satisfy the non-negative constraint given in section 7.2.8 below.

7.2.6. Land Constraints

The sum of the amount of land allocated for each crop j, at stage i, on plot k, must be less than or equal to the total area of land allocated for crop production at stage i, on plot k. This constraint is given by equation 7.2 below.

$$\sum_{j}^{M_{ki}} X_{kij} \leq L_{ki} \quad \forall k, i \tag{7.2}$$

Feasible solutions must satisfy the lower and upper bound constraints. This will ensure that the feasible solutions found will be relative to the market demand in view of the current agricultural practices. This constraint is given by equation 7.3 below.

$$Lb_{kij} \leq X_{kij} \leq Ub_{kij} \quad \forall k, i, j \tag{7.3}$$

7.2.7. Irrigation Constraints

The total volume of irrigated water required for the production of all crops, within a year, must be less than or equal to the total volume of irrigated water that can be supplied to the given area of land. This constraint considers that some crops may require more irrigated water than what is supplied ha^{-1}. It is therefore the responsibility of the farmer to distribute his supply of irrigated water efficiently. This constraint is given by equation 7.4 below.

$$\sum_{k} \sum_{i} \sum_{j} IR_{kij} \leq TA \tag{7.4}$$

7.2.8. Non-negative Constraints

The gross profits that can be earned per crop must be greater than zero. This constraint is given by equation 7.5.

$$B_{kij} > 0 \quad \forall k, i, j \tag{7.5}$$

7.3 Case Study of the Vaalharts Irrigation Scheme

The statistics for the primary crops grown in this area is given in Table 7.3.1 (Maisela, 2010). These statistics have been determined over a 5 year period. It includes the hectares allocated per crop (ha's crop^{-1}) and the average tons of returns produced per hectare per crop (t ha^{-1}). The crops grown all year round consist of the perennial (p) crops. These crops grow on the single-crop plots of land. The seasonal crops consist of the summer and winter crops. These are primarily grown on the double-crop plots of land.

With the current agricultural practices, the total area of land allocated for the cultivation of perennial crops is calculated to be 8,300 ha. The total area of land allocated for the cultivation of the summer crops is 15,500 ha. The land allocated for the cultivation of the winter crops is 12,200 ha.

The Crop Water Requirement (CWR) for each crop is provided by Brouwer and Heibloem (1986). The average rainfall for the growing months of each crop is determined from Maisela (2010). The producer prices of a ton of yield produced from each crop (ZAR[1] t^{-1}) is given by the Directorate Statistics and Economic Analysis (2012).

[1] ZAR stands for Zuid-Afrikaanse Rand. It is the Dutch translation of saying, "South African Rand." The Rand is the currency in South Africa.

Table 7.3.1: Vaalharts Irrigation Scheme crop and average rainfall statistics

Crops	ha's crop^{-1}	t ha^{-1}	CWR	AR	ZAR t^{-1}
Pecan Nuts (p)	100	5.0	1,600	444.7	3,500.00
Wine Grapes (p)	300	9.5	850	350.8	2,010.00
Olives (p)	400	6.0	1,200	444.7	2,500.00
Lucerne (p)	7,500	16.0	1,445	444.7	1,185.52
Cotton (s)	2,000	3.5	700	386.4	4,500.00
Maize (s)	6,500	9.0	979	279.0	1,321.25
Ground Nuts(s)	7,000	3.0	912	339.5	5,076.00
Barley (w)	200	6.0	530	58.3	2,083.27
Wheat (w)	12,000	6.0	650	58.3	2,174.64

7.4 Testing and Evaluation

The non-heuristic specific parameters, required for the execution of the algorithms, had been set according to the values given in Table 7.4.1. The lower and upper bounds ensure that feasible solutions are found which relate to the current agricultural practices of the irrigation scheme. $F_{kij} \in [0,1]$. C_IR_{kij} is the cost of irrigated water (ZAR ha^{-1}). O_{kij} is set to a third of the producer prices per ton of yield (ZAR ha^{-1}). These values are sufficient to evaluate the performances of the metaheuristic algorithms, in comparing them to the results of the current agricultural practices.

Table 7.4.1: Non-heuristic specific parameters required for the execution of the algorithms

Crops	Lb_{kij}	Ub_{kij}	F_{kij}	C_IR_{kij}	O_{kij}
Pecan Nuts	50	150	1	1,013.20	5,833.35
Wine Grapes	150	450	1	437.80	6,365.00
Olives	200	600	1	662.40	4,999.98
Lucerne	7,100	7,900	1	877.26	6,322.72
Cotton	1,000	3,000	1	275.03	5,250.00
Maize	5,000	8,000	1	613.90	3,963.78
Ground Nuts	4,500	9,500	1	502.08	5,076.00
Barley	100	300	1	413.68	4,166.52
Wheat	11,900	12,100	1	518.92	4,349.28

The initial parameter settings for the LS metaheuristic algorithms were set as follow:

- BPA – The $listSize$ was set at 20. The $noOfIterations$ was set at 20,000. p_a was set at 0.2.

- IBPA – The $listSize$ was set at 20. The $noOfIterations$ was set at 1,000. The $stepsPerChange$ was set at 20.
- LADA – The $listSize$ was set at 20. The $noOfIterations$ was set at 10,000. m was set at 2.
- TS – The $tabuListSize$ was set at 7. The $candidateListSize$ was set at 20. The $noOfIterations$ was set at 1,000.
- SA – The $stepsPerChange$ was set at 50. T was set at 50. F was set at 0.9. α was set at 0.99.

The initial parameter settings for the population based metaheuristic algorithms were set as follows:

- CS – The number of host bird nests n was set at 20. The $noOfIterations$ was set at 20,000. p_a was set at 0.25.
- FA – The number of fireflies n was set at 20. The $noOfIterations$ was set at 1,000. α was set at 0.25, β_0 at 0.2 and γ at 1.
- GSO – The number of glow-worms n was set at 20. The $noOfIterations$ was set at 1,000. l_0 was set at 1, r_0 at 1.2, r_s at 1.5, ρ at 0.4, γ at 0.6, β at 0.08, st at 0.3 and N_d at 10.
- GA – The number of individuals n was set at 20. The $maxNoOfGenerations$ was set at 1,000. $cRate$ was set at 0.8. $mRate$ was set at 0.05 $(1/n)$.

The $tabuListSize$ of TS was set according to the recommended settings given by Sarmady (2012). For SA, α was set high enough to ensure a slow annealing process. For CS, p_a was set according to the setting given in Xin-She Yangs' implementation of CS (Yang, 2010). α, β_0 and γ were set according to the settings given in Xin-She Yangs' implementation of an m-dimensional Firefly Algorithm (Yang, 2010). For GSO, ρ, γ, β and st were set according to the settings given in (Zhao *et al.*, 2012). l_0, r_0 and r_s are problem specific parameters. N_d was set to half of the number of glow-worms n. For GA, $cRate$ was set at 0.8. This value was used after several tests had been performed to determine the best probable crossover rate to implement.

To compare the metaheuristic algorithms fairly, the list sizes of BPA, IBPA and LADA, the candidate list size of TS and the 'population' sizes for CS, FA, GSO and GA were all set to be the same, i.e. $n = 20$. The $noOfIterations$ of BPA, IBPA, LADA, TS, CS, FA and GSO, the $maxNoOfGenerations$ of GA and the parameter settings of SA ensured that each algorithm executed for 20,000 objective function evaluations. 20,000 objective function evaluations are sufficient to compare the performances of the metaheuristic algorithms for this dataset of 9 crops, given the lower and upper bound settings. For larger datasets the complexity of the problem will increase exponentially. For such instances a larger number of objective function evaluations will be needed. Each algorithm was run 100 times, using randomly generated population sets for each run.

To ensure fairness, the 100 different population sets had been initially randomly generated. Each population set contained 20 solutions. Mathematically, the study denotes a population set as pop_i, for $i = 1, \dots, 100$. Then, for each run i, pop_i was used as an input parameter for BPA, IBPA, LADA, CS, FA, GSO and GA. For the LS algorithms, this was to set the Performance List's (PL's) for BPA and IBPA and the Solutions List (SL) for LADA. For the population based metaheuristics, this was to set the different populations for each algorithm. This means that for run $i = 1$; BPA, IBPA, LADA, CS, FA, GSO and GA was run using pop_1, for run $i = 2$; BPA, IBPA, LADA, CS, FA, GSO and GA was run using pop_2, and so on until $i = 100$. For each population set pop_i, the best solution from each set was also used to initialize "best" for TS and SA.

From the 100 best solutions determined by each metaheuristic algorithm, the results of the best and average solutions have been documented. Using the population of the 100 best solutions determined by each metaheuristic algorithm, the 95% confidence interval[2] values have been calculated for the execution times and for the fitness values (total gross profits earned). The results are explained below.

[2] In statistics a confidence interval (CI) indicates the reliability of an interval estimate of population parameters. 95% CI means to be 95% certain that the population parameters will lie within the interval estimate range.

Table 7.4.2 give the statistics of the average execution times (AVG) in milliseconds (ms), and the 95% confidence interval (95% CI) values of each metaheuristic algorithm.

Table 7.4.2: The average execution times, in milliseconds, and the 95% CI values of each metaheuristic algorithm

Methods	**AVG (ms)**	**95% CI**
BPA	38	AVG $\pm$ 0.4
IBPA	36	AVG $\pm$ 0.8
LADA	23	AVG $\pm$ 0.4
TS	33	AVG $\pm$ 0.3
SA	36	AVG $\pm$ 0.6
CS	147	AVG $\pm$ 1.1
FA	444	AVG $\pm$ 6.4
GSO	126	AVG $\pm$ 2.7
GA	141	AVG $\pm$ 2.4

From Table 7.4.2 it can be observed that LADA executed the fastest overall. The average execution times of BPA, IBPA, TS and SA are all comparable. The relatively fast execution time of LADA is due to its ability to work with two solutions per iteration.

The average execution times of the population based metaheuristic algorithms were much slower. FA took the longest time to execute overall. The average execution times of CS, GSO and GA were all comparable. The relatively large average execution time of FA is due to its nested *for* loop. In this *for* loop, each firefly's fitness value is compared to the fitness value of every other firefly. This has shown to be computationally expensive.

For the population based algorithms, the execution time of GSO is the fastest. This is due to the limitation on the maximum number of neighbours that a glow-worm is allowed to have. As the number of iterations increase, the vision ranges of the glow-worms will decrease. This will cause the glow-worms to become more separated in searching the local neighbourhood structures of the solution space. This separation will reduce the number of glow-worms considered in searching for neighbours, which will speed up the execution process.

The 95% CI values, from Table 7.4.2, indicates that one can be 95% certain that the 100 execution times of each algorithm have fallen within those interval estimates. By observing the CI values, one can conclude that the execution times of the algorithms have been fairly consistent. A visual representation of the statistical values given in Table 7.4.2 is shown in Figure 7.4.1 below. In Figure 7.4.1, the 95% CI values are represented by the black interval estimates.

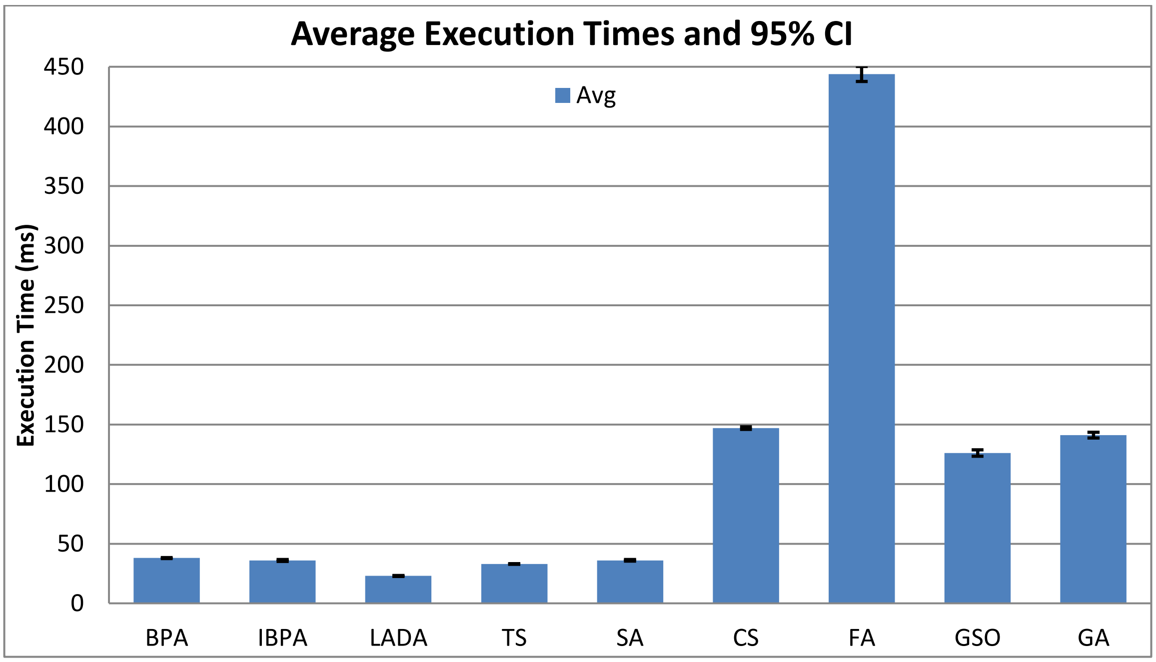

Figure 7.4.1: The average execution times, in milliseconds (ms), and the 95% CI values of each metaheuristic algorithm

Table 7.4.3 gives the statistical values of the overall best (BFV) and average best (ABFV) fitness values of each metaheuristic algorithm. The fitness values are the total gross profits earned. The 95% CI values for the fitness value populations of each algorithm are also given, along with the current agricultural practice (CP) earning.

Table 7.4.3: Statistics for the best fitness values (BFV), average best fitness values (ABFV) and 95% CI values

Methods	BFV (ZAR)	ABFV (ZAR)	95% CI
CP	332,027,707	N/A	N/A
BPA	336,460,533	336,448,988	ABFV $\pm$ 1,068.1
IBPA	336,459,139	336,450,788	ABFV $\pm$ 892.4
LADA	336,453,273	336,436,712	ABFV $\pm$ 1,752.1
TS	336,456,927	336,441,272	ABFV $\pm$ 1,586.9
SA	336,249,577	335,887,711	ABFV $\pm$ 23,782.4
CS	336,461,787	336,459,391	ABFV $\pm$ 255.0
FA	336,366,886	336,119,823	ABFV $\pm$ 37,178.5
GSO	336,419,655	335,745,122	ABFV $\pm$ 136,260.2
GA	336,219,977	335,813,775	ABFV $\pm$ 34,344.2

From Table 7.4.3, it is observed that each of the metaheuristic algorithms have determined an overall BFV that is superior to the CP at the irrigation scheme.

Of all the algorithms, CS determined the highest BFV. This is followed by BPA, IBPA, TS, LADA, GSO, FA, SA and then GA. CS determined a best solution that earned an extra gross profit of ZAR 4,434,080. BPAs' best solution earned an extra gross profit of ZAR 4,432,826. The best solutions of IBPA, TS, LADA, GSO, FA, SA and GA earned extra gross profits of ZAR 4,431,432, ZAR 4,429,220, ZAR 4,425,566, ZAR 4,391,948, ZAR 4,339,179, ZAR 4,221,870 and ZAR 4,192,270 respectively. On average, the gross profits earned by each of the metaheuristic algorithms were also higher than that of the CP. CS had the highest ABFV. This was followed by IBPA, BPA, TS, LADA, FA, SA, GSO and then GA.

A graphical comparison of the algorithm's best and average fitness values, as determined from Table 7.4.3, is shown in Figure 7.4.2. The 95% CI values are represented by the black interval estimates over the range of average fitness values.

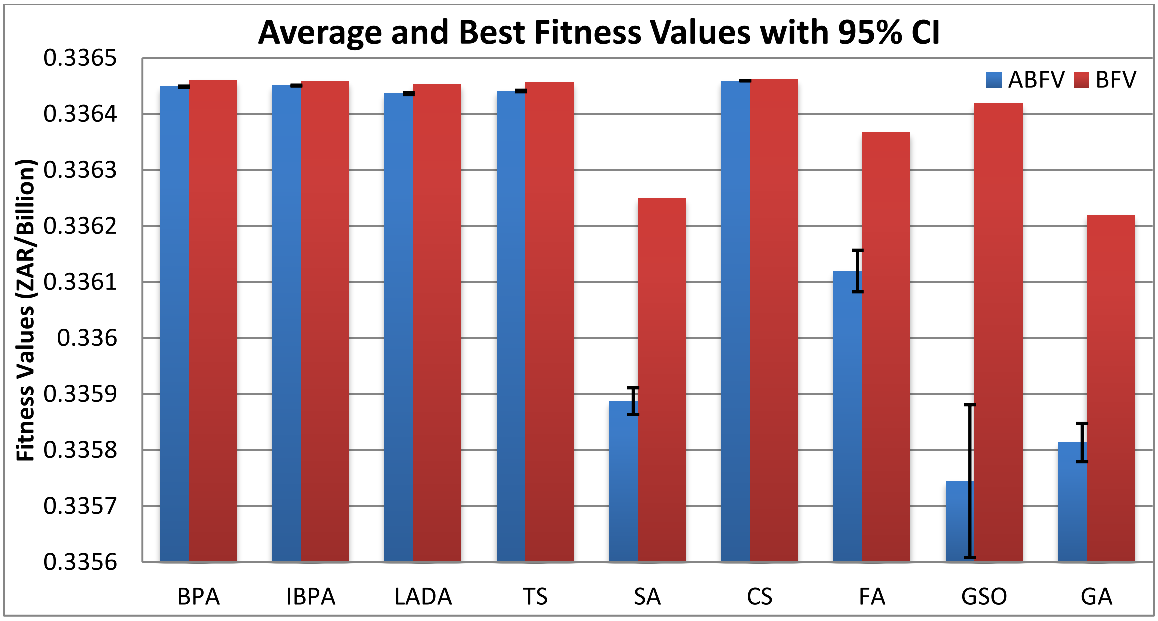

Figure 7.4.2**:** A comparison of each algorithms best and average fitness values determined, along with the 95% CI estimates

For the LS metaheuristic algorithms, Figure 7.4.2 shows that the differences between the ABFV's and the differences between the BFV's of BPA, IBPA, LADA and TS are minimal. The ABFV and BFV of SA are relatively inferior to the values of the other LS algorithms. For the population based algorithms, CS delivered the highest ABFV performance. This is followed by FA, GA and then GSO.

The 95% CI value of CS is also the least overall. This is followed by IBPA, BPA, TS, LADA, SA, GA, FA and GSO. Having determined the overall best and average fitness values, and with CS having the lowest 95% CI value proves that it has been the strongest and most consistent metaheuristic algorithm for this particular optimization problem. For the LS metaheuristic algorithms, by observing the fitness value performances and their 95% CI fitness values, one can conclude that BPA and IBPA have been the strongest algorithms. SA, FA and GA have similar 95% CI values. GSO has the largest 95% CI value overall. For the LS algorithms it is concluded that TS performed better than LADA. SA performed the worst. For the population based algorithms, it is concluded that FA performed better than GSO and GA. GA performed the worst overall.

For the LS algorithms, the strength of BPA, IBPA and LADA following from their performances, is attributed to their ability to maintain updated lists of their best solutions found. Maintaining updated lists allow the algorithm's to work with a limited number of their best solutions found. Working with multiple best solutions allows for exploration of the solution space. Performing local search facilitates exploitation within the local neighbourhood structures of the solution space. The performances of these three algorithms prove that they have good balances in performing exploration and exploitation of the solution space for this particular optimization problem.

The ability to maintain updated lists of their best solutions is the primary difference in the performances of the new algorithm's compared to TS and SA. With TS several possibly good solutions don't get exploited due to the fact that only a single solution is being selected from the candidate list at each iteration. This technique means the TS is strong in exploitation, but lacks slightly in exploration. With SA, the ability to accept worse solutions is the reason for its relatively poor performance. Accepting worse solutions facilitates exploration of the solution space. This particular optimization problem however requires that the LS algorithms have stronger exploitation abilities. This is the primary reason why TS has performed better than SA.

For the population based algorithms, the strength of CS lies in its ability to improve on the population of host bird nest solutions, in using the best nest solution from the previous iteration. The best nest solution is used in performing levy flights. If solutions are found which improve on the host bird nest solutions, then the inferior host bird nest solutions will get replaced by the improved solutions in moving closer to the best nest solution. This results in a population of host bird nest solutions that have found the most promising areas within the domains of the solution space. Worst solutions are not considered in performing exploitation in using this technique. The probability of the host bird discovering intrusions facilitates exploration. The best host bird nest solution found will then be used to direct the search in the next iteration. From all population based algorithms, CS seems to have the best balance in exploring and exploiting the local neighbourhood structures of the solution space, for this optimization problem.

GSO delivered the worst average performance and has the highest 95% CI fitness value. This is due to its weakness in exploitation of the local neighbourhood structures of the solution space. As the iteration count increases, a reduction in the glow-worms vision ranges causes group-like separations of the glow-worms throughout the domains of the solution space. This technique encourages exploration but discourages exploitation. These separations result in fewer glow-worms searching the local neighbourhood structures of the solution space. A glow-worm moving towards another glow-worm (with a higher level of luciferin than itself) may also accept a worse solution. These two factors will not result in the most effective exploitation of a solution space, on average. GSOs' high best fitness value and high 95% CI fitness value proves that GSO had determined good, but also poor solutions.

Similar to GSO, the fireflies in FA also accept worse solutions while moving towards brighter fireflies. However, they do not deliberately cause group-like separations throughout the domains of the solution space. This allows for better exploitation, compared to GSO. The main difference in the performances of CS and FA still lie in FAs' ability to accept worse solutions.

Figure 7.4.3 shows the performances of the heuristic algorithms, in determining their BFV solutions. It is observed that all algorithms quickly determined improved solutions over the current agricultural practices (CP). For the LS algorithms, BPA, IBPA and TS performed very similarly in determining their best overall solutions. LADA also performed similarly to these algorithms, but it can be seen that its performance was slightly inferior. SA progressed at a much slower rate. SA found its best solution at around 11,000 objective function evaluations.

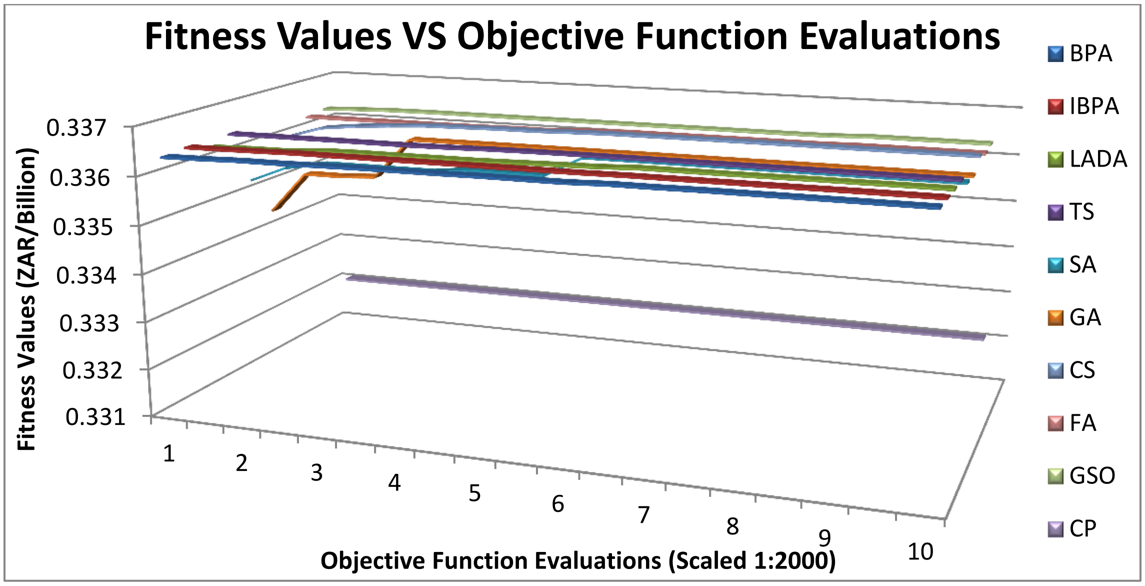

Figure 7.4.3: The performance of the metaheuristic algorithms in determining their overall best fitness values

For the population based algorithms, GA found its best solution at around 6000 objective function evaluations. CS, GSO and FA had already found similar solutions to GAs' best solution at around 4,000 objective function evaluations. From this point onwards CS, GSO and FA showed slight improvements. CS determined its best solution at around 10,000 objective function evaluations.

Table 7.4.4: Statistics of the irrigated water requirements (IWR) and variable costs of production (VCP) values for the best solutions found

Methods	IWR (m^3)	VCP (ZAR)
CP	244,491,000	198,176,322
BPA	241,099,517	199,946,516
IBPA	241,090,140	199,944,586
LADA	241,092,160	199,941,918
TS	241,084,342	199,942,717
SA	241,101,715	199,841,332
CS	241,084,702	199,945,194
FA	241,077,145	199,896,752
GSO	241,058,226	199,920,593
GA	241,220,612	199,842,172

Table 7.4.4 gives the statistics of the irrigated water requirements (IWR) and the variable costs of production (VCP) values for the best solution determined by each of the metaheuristic algorithms. It also gives the statistics at the current agricultural practices (CP).

As can be seen from Table 7.4.4, each metaheuristic algorithm determined a best solution that required reduced volumes of irrigated water, compared to CP. GSO found a solution that required the least volume of irrigated water. This is followed by FA, TS, CS, IBPA, LADA, BPA, SA and then GA. GSOs' solution saved a total volume of 3,432,774 m^3. FAs' solution saved a total volume of 3,413,855 m^3. The volume of irrigated water saved by TS, CS, IBPA, LADA, BPA, SA and GA was 3,406,658 m^3, 3,406,298 m^3, 3,400,860 m^3, 3,398,840 m^3, 3,391,483 m^3, 3,389,285 m^3 and 3,270,388 m^3 respectively. At the quota of 9,140 $m^3ha^{-1}annum^{-1}$, the savings determined by GSO, FA, TS, CS, IBPA, LADA, BPA, SA and GA would be able to supply irrigated water to an extra 375, 373, 373, 372, 372, 372, 371, 371 and 357 hectares of agricultural land respectively.

From Table 7.4.4, the relative increases in the VCP values of each algorithm, compared to CP, is acceptable considering the increased total gross profits earned.

A graphical representation of the IWR values, as determined from Table 7.4.4, is shown in Figure 7.4.4.

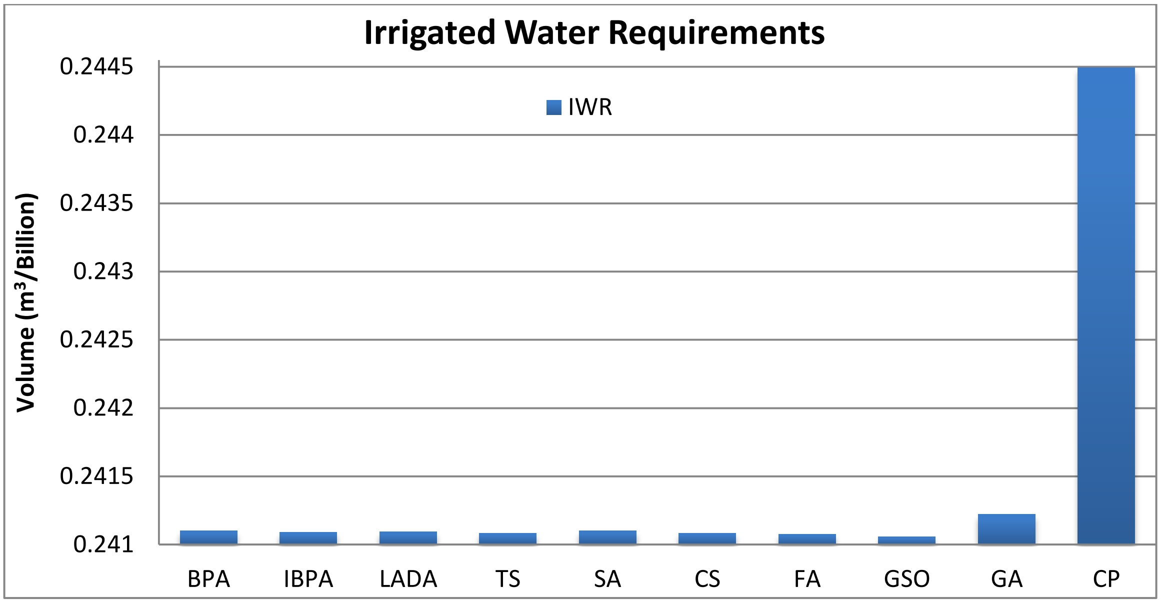

Figure 7.4.4: Irrigated water requirements (IWR) of the current agricultural practices (CP) and those of the best metaheuristic solutions

Figure 7.4.5 gives a graphical comparison of the seasonal hectare allocations of each crop, at the current agricultural practices (CP) and that of the best solution determined by each metaheuristic algorithm.

As can be observed from Figure 7.4.5, each metaheuristic algorithm determined that primarily increasing the hectare allocations for cotton and ground nuts and decreasing the hectare allocations for maize were the main differences in determining improved solutions. The higher producer prices t^{-1} and lower irrigated water requirements ha^{-1} of cotton and ground nuts, result in higher profits ha^{-1} being earned for those two crops. This, coupled with the reduction in the hectare allocations for maize, resulted in higher profits being earned for the given area of land.

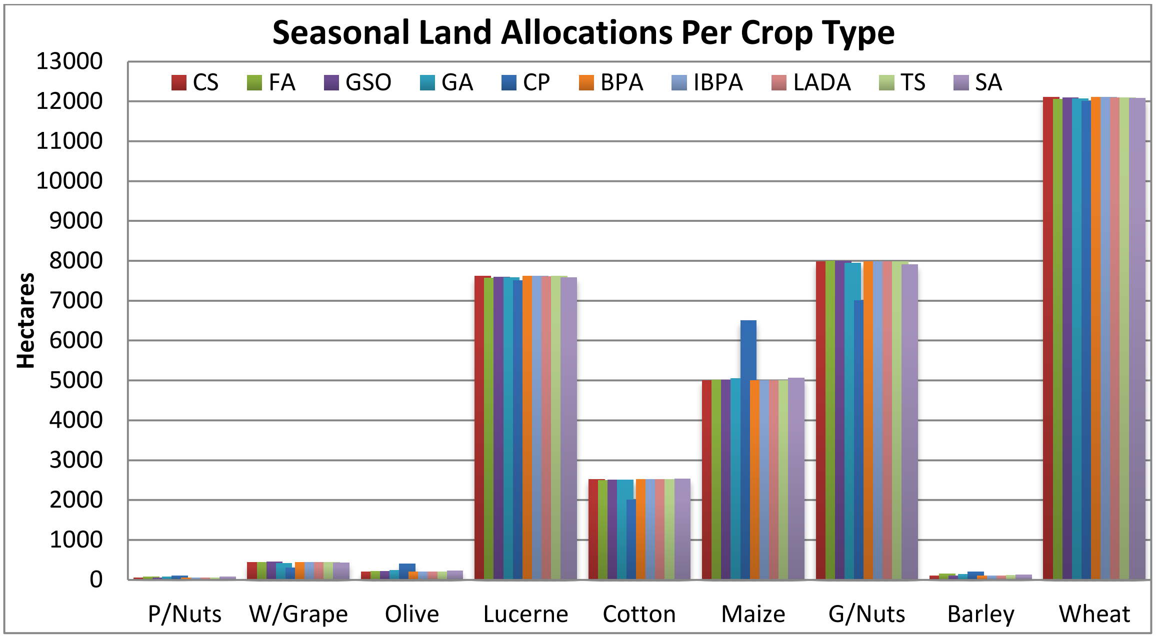

Figure 7.4.5: A comparison of the hectare allocations, per crop, at the current agricultural practices and for the best metaheuristic solutions

Table 7.4.5, 7.4.6 and 7.4.7 give the statistical values of each crop's hectare allocations (ha's crop^{-1}), irrigated water requirements (IWR) and variable costs of production (VCP) at the current agricultural practices (CP), and that of the best solution determined by each metaheuristic algorithm.

The program was written with the Java programming language. It was programmed using the Netbeans® 7.0 Integrated Development Environment. All simulations were run on the same platform. The computer used had a Windows® 7 Enterprise operating system, an Intel® Celeron® Processor 430, 3 GB of RAM and a 500GB hard-drive.

Table 7.4.5: Statistics of the current agricultural practices (CP) and metaheuristic solutions per crop

Crops	Methods	ha's crop^{-1}	IWR (m^3)	VCP (ZAR)
Pecan Nuts	CP	100	1,155,300	684,655
	BPA	50	581,875	344,831
	IBPA	50	582,499	345,201
	LADA	51	593,125	351,498
	TS	51	586,361	347,489
	SA	69	792,894	469,885
	CS	50	579,742	343,567
	FA	78	903,255	535,288
	GSO	53	616,177	365,160
	GA	77	888,848	526,750
Wine Grapes	CP	300	1,497,600	2,040,840
	BPA	432	2,154,918	2,936,593
	IBPA	434	2,166,747	2,952,712
	LADA	434	2,166,095	2,951,825
	TS	434	2,164,291	2,949,366
	SA	430	2,146,469	2,925,080
	CS	434	2,167,416	2,953,625
	FA	442	2,208,255	3,009,278
	GSO	449	2,241,026	3,053,936
	GA	414	2,065,137	2,814,244
Olives	CP	400	3,021,200	2,264,951
	BPA	203	1,530,353	1,147,284
	IBPA	203	1,531,019	1,147,784
	LADA	204	1,539,091	1,153,835
	TS	203	1,535,232	1,150,942
	SA	223	1,684,414	1,262,782
	CS	203	1,531,624	1,148,237
	FA	212	1,600,413	1,199,807
	GSO	210	1,585,648	1,188,738
	GA	231	1,747,699	1,310,225
Lucerne	CP	7,500	75,022,500	53,999,873
	BPA	7,615	76,176,295	54,830,354
	IBPA	7,613	76,151,169	54,812,269
	LADA	7,611	76,132,584	54,798,892
	TS	7,612	76,147,166	54,809,388
	SA	7,578	75,806,480	54,564,169
	CS	7,613	76,151,414	54,812,446
	FA	7,568	75,698,368	54,486,351
	GSO	7,588	75,900,819	54,632,072
	GA	7,578	75,802,563	54,561,349

Table 7.4.6: Statistics of the current agricultural practices (CP) and metaheuristic solutions per crop

Crops	Methods	ha's crop^{-1}	IWR (m^3)	VCP (ZAR)
Cotton	CP	2,000	6,272,000	11,050,054
	BPA	2,520	7,901,862	13,921,557
	IBPA	2,519	7,900,705	13,919,518
	LADA	2,518	7,895,479	13,910,312
	TS	2,520	7,901,970	13,921,747
	SA	2,535	7,951,260	14,008,586
	CS	2,520	7,902,298	13,922,324
	FA	2,498	7,834,265	13,802,464
	GSO	2,500	7,840,663	13,813,737
	GA	2,504	7,851,488	13,832,807
Maize	CP	6,500	45,500,000	29,754,920
	BPA	5,000	35,001,705	22,889,515
	IBPA	5,001	35,008,433	22,893,914
	LADA	5,002	35,010,674	22,895,380
	TS	5,001	35,005,960	22,892,298
	SA	5,062	35,434,009	23,172,222
	CS	5,000	35,000,825	22,888,939
	FA	5,009	35,064,728	22,930,729
	GSO	5,008	35,053,734	22,923,540
	GA	5,049	35,340,296	23,110,938
Ground Nuts	CP	7,000	40,075,000	39,046,578
	BPA	7,980	45,685,672	44,513,266
	IBPA	7,979	45,682,283	44,509,964
	LADA	7,981	45,689,989	44,517,472
	TS	7,979	45,681,995	44,509,683
	SA	7,903	45,241,930	44,080,911
	CS	7,980	45,685,596	44,513,192
	FA	7,993	45,757,531	44,583,281
	GSO	7,992	45,754,842	44,580,661
	GA	7,948	45,500,715	44,333,055
Barley	CP	200	943,400	916,040
	BPA	100	473,907	460,163
	IBPA	100	472,145	458,452
	LADA	102	480,650	466,711
	TS	105	495,415	481,048
	SA	119	562,667	546,349
	CS	101	478,041	464,177
	FA	148	696,036	675,850
	GSO	102	479,890	465,972
	GA	136	642,818	624,175

Table 7.4.7: Statistics of the current agricultural practices (CP) and metaheuristic solutions per crop

Crops	Methods	ha's crop^{-1}	IWR (m^3)	VCP (ZAR)
Wheat	CP	12,000	71,004,000	58,418,411
	BPA	12,100	71,592,931	58,902,953
	IBPA	12,100	71,595,142	58,904,772
	LADA	12,098	71,584,473	58,895,994
	TS	12,095	71,565,952	58,880,755
	SA	12,081	71,481,592	58,811,348
	CS	12,099	71,587,746	58,898,687
	FA	12,052	71,314,293	58,673,704
	GSO	12,098	71,585,427	58,896,779
	GA	12,064	71,381,050	58,728,628

7.5 Conclusion

This chapter addresses an Annual Crop Planning (ACP) problem at the Vaalharts Irrigation Scheme (VIS), in South Africa. Due to increase in costs associated with agricultural production, and limited availability of resources, it is important that efficient solutions be found to the problem of ACP.

To determine efficient solutions, this chapter presents a new ACP mathematical model. This model is intended to determine ACP solutions at *existing* irrigation schemes. The objective function aims at making efficient use of the limited resources available in maximizing total gross profits. The limited resources include land, irrigated water supply and the variable costs associated with agricultural production.

To determine solutions to the case study problem of this NP-Hard optimization problem, three new Local Search (LS) and three relatively new Swarm Intelligence (SI) metaheuristic algorithms have been investigated. The LS algorithms include the Best Performance Algorithm (BPA), the Iterative Best Performance Algorithm (IBPA) and the Largest Absolute Difference Algorithm (LADA). The SI algorithms include Cuckoo Search (CS), Firefly Algorithm (FA) and Glowworm Swarm Optimization (GSO). To determine the relative merits of the solutions found, the solutions of the LS algorithms have been compared with the solutions determined by Tabu Search (TS) and Simulated Annealing (SA). To determine the relative merits of the solutions found by the SI algorithms, their solutions have been compared with the solutions determined by the Genetic Algorithm (GA). All metaheuristic solutions have also been compared to one another and to the statistics of the current agricultural practices at the VIS.

To ensure fairness in the performances of the metaheuristic algorithms, the algorithm specific parameter settings of TS, CS, FA and GSO had been set according to recommended settings. Other parameter settings, such as the 'list' sizes, the 'population' sizes and the initial population sets were set to be the same. The parameter settings ensure that the total number of objective function evaluations, per run, would be the same for each algorithm. Each metaheuristic algorithm was run 100 times. From these 100 runs the overall best and average solutions of each algorithm were documented.

From the solutions documented, one can observe that each metaheuristic algorithm provides superior solutions to that of the current agricultural practice (CP). Each algorithm's overall best solution determined seasonal hectare allocations that increased gross profits and reduced the irrigated water requirements. Each algorithm determined that primarily increasing the hectare allocations for cotton and ground nuts, and decreasing the hectare allocations for maize were the main differences in determining improved solutions over the current agricultural practices.

CS delivered the best comparative solutions compared to other methods. It delivered the best overall solution, was the best on average, and had the lowest 95% confidence interval fitness value. This proved that CS had consistently found very good areas within the domains of the solution space. It is concluded that CS is the best metaheuristic algorithm for this particular optimization problem. For the LS algorithms, BPA and IBPA were the best performers. Of all the metaheuristic algorithms, GA performed the worst.

CSs' strength is attributed to its balance in exploring and exploiting the local neighbourhood structures of the solution space. The strength of the new LS algorithms, in their performance, is attributed to their ability to maintain updated lists of their best solutions found. Maintaining updated lists allows these algorithms to work with multiple best solutions in exploring the solution space.

It has also been observed that the GSO has the ability to determine very good solutions, but due to the weakness in its exploitation ability, it has performed the worst on average. For the LS algorithms, TS performed better than LADA, while SA was the worst. For the population based algorithms, FA performed better than GSO and GA, while GA performed the worst overall.

CHAPTER EIGHT
PERFORMANCE ON A NEW IRRIGATION SCHEME

8.1 Introduction

The increase in population has increased the need for more food to be produced. Currently, the lack of food supply and the increase in producer prices have resulted in a large percentage of people being unable to afford sufficient food. According to the Food and Agriculture Organization of the United Nations, it is now estimated that more than a billion people suffer from undernourishment (FAO, 2010). The problems of hunger and starvation are particularly predominant in the developing countries of this world. To try to combat the problem of a lack of food supply and increased producer prices, it is important that more land be made available for agricultural production and that optimized solutions be found to crop planning problems.

For more food to be produced it is necessary that the agricultural sector increase its output. This is because the agricultural sector is the primary supplier of food in the world (Schmitz *et al.*, 2007). As discussed in chapter seven, determining optimized solutions to crop planning is important, but not sufficient to meet the future demands of food. To produce more food for the future, more land must be made available for agricultural production.

This chapter presents the problem of Annual Crop Planning (ACP) at *new* irrigation schemes. Once land is allocated for the development of a new irrigation scheme, and the crops to be produced have been finalized, then optimized solutions will need to be found regarding resource allocations amongst the various competing crops that are to be grown. To determine optimized solutions, an ACP mathematical model is formulated and presented in this chapter. This model is similar to the model presented in chapter seven, with slight but significant differences. This model is intended to determine solutions to the ACP problem at new irrigation schemes. The functions' objective is also to maximize total gross profits in efficiently allocating the limited resources available for crop production, within a production year.

To determine solutions to the ACP problem for a new irrigation scheme, this chapter investigates the abilities of the Local Search (LS) and population based metaheuristic algorithms presented in chapters five and six. The LS algorithms are the Best Performance Algorithm (BPA), the Iterative Best Performance Algorithm (IBPA), the Largest Absolute Difference Algorithm (LADA), Tabu Search (TS) and Simulated Annealing (SA). The population based algorithms include Cuckoo Search (CS), the Firefly Algorithm (FA), Glowworm Swarm Optimization (GSO) and the Genetic Algorithm (GA). These algorithms will be compared in their ability to determine solutions to the ACP case study problem for a new irrigation scheme. The solutions found will be valuable in making suggesting concerning the seasonal hectare allocations of the crops that are required to be grown at a new irrigation scheme.

8.2 New Irrigation Schemes

Unless a portion of land is privately owned, and there are infrastructures available for irrigated water supply, the land made available for the development of new irrigation schemes is allocated by the government. The government will do so when the need arises and for the sake of social and economic development. For land to be made available for the development of new irrigation schemes, it needs to be assessed to determine its feasibility for crop production. There are several factors that need to be considered in determining the feasibility of a portion of land. Some of the most important factors include the soil conditions, the climatic conditions, the availability of natural resources, the sustainability of crop production and agricultural trends, amongst others. The sustainability of crop production would determine the future success of the irrigation scheme.

The nutritional value of the soil, its field capacity and its natural drainage system are important factors in determining the suitability of the crops given the soil conditions. The climatic conditions also determine the types of crops that will be most suitable. Considering the availability of natural land resources is also important. Natural land resources such as lakes and rivers are very valuable because the natural land resources can be used to source irrigated water. Irrigated water and rainfall are important in determining the full agricultural potential of a given area of land. The agricultural trends determine the types of crops that will be most suitable for economic benefits.

Central to the construction of an irrigation scheme is the infrastructure to transport irrigated water. If no infrastructure exists then a system will need to be developed to transport the irrigated water. The type of infrastructure built will depend on the irrigated water needs of the given area of land. The development of an irrigation system is very costly. It is therefore important that the crops selected to be grown should be profitable enough to meet the financial investments involved in the development of the irrigation scheme. The construction of natural or artificial drainage systems is also important.

When an area of land gets allocated for the development of a new irrigation scheme, and it has been finalized which crops will be produced, then solutions need to be found concerning the hectare allocations amongst the various crops that are to be grown. In determining the hectare allocations, the planting and harvesting schedules of the different types of crops must be considered. However, in order to do so, the hectare allocations of the different farm plot types need to be considered first. The problem of trying to optimize the seasonal hectare allocations amongst the various competing crops that are to be grown within the year is therefore an ACP problem. In addition to determining the hectare allocations amongst the various competing crops, the hectare allocations for the farm plot types will also need to be determined. Feasible solutions found must satisfy the multiple hard and soft constraints that are associated with ACP for a new irrigation scheme.

8.3 The ACP Model for a New Irrigation Scheme

This model is similar to the model presented in chapter seven, with the additional complexity of determining solutions to the farm plot sizes for the different farm plot types. The functions' objective is the same, which is to maximize gross profits in efficiently allocating the limited resources amongst the various competing crops required to be grown within a production year. Similarly, feasible solutions must satisfy all constraints associated with this problem.

8.3.1 Indices

- k – Plot types. (1 = single-crop plots, 2 = double-crop plots, 3 = triple-crop plots, and so on).
- i – Indicative of the groups of crops that are grown in sequence throughout the year, on plot type k ($i = 1$ represents the 1st group of sequential crops, $i = 2$ represents the 2nd group of sequential crops, $i = 3$ represents the 3rd group of sequential crops, and so on).
- j – Indicative of the individual crops grown at stage i, on plot k.

8.3.2 Input Parameters

- l – Number of different plot types.
- N_k – Number of sequential groups of crops grown within a year, on plot k.
- M_{ki} – Number of different types of crops grown at stage i, on plot k.
- F_{kij} – Average fraction per hectare of crop j, at stage i, on plot k, which needs to be irrigated (1 = 100% coverage, 0 = 0% coverage).
- R_{kij} – Averaged rainfall estimates that fall during the growing months for crop j, at stage i, on plot k.
- CWR_{kij} – Crop Water Requirements of crop j, at stage i, on plot k.
- T – Total hectares of land allocated for the irrigation scheme.
- A – Volume of irrigated water that can be supplied per hectare (ha^{-1}).
- P – Price of irrigated water m^{-3}.
- O_{kij} – Other operational costs ha^{-1} of crop j, at stage i, on plot k. These costs exclude the cost of irrigation.
- YR_{kij} – The amount of yield that can be obtained in tons per hectare ($t\ ha^{-1}$) from crop j, at stage i, on plot k.
- MP_{kij} – Producer prices per ton (t^{-1}) for crop j, at stage i, on plot k.
- Lb_{kij} – Lower-bound for crop j, at stage i, on plot k.
- Ub_{kij} – Upper-bound for crop j, at stage i, on plot k.
- Lb_P_k – Lower-bound for plot type k.
- Ub_P_k – Upper-bound for plot type k.

8.3.3 Calculated Parameters

- IR_{kij} – Volume of irrigated water estimates that should be applied to crop j, at stage i, on plot k. ($IR_{kij} m^3 = (CWR_{kij} m - R_{kij} m) * 10000 m^2 * F_{kij}$).
- TA – Total volume of irrigated water that can be supplied to the given area of land, within a year ($TA = T * A$).
- C_IR_{kij} – The cost of irrigated water ha^{-1} of crop j, at stage i, on plot k. ($C\_IR_{kij} = IR_{kij} * P$).
- C_{kij} – Variable costs ha^{-1} of crop j, at stage i, on plot k. ($C_{kij} = O_{kij} + C\_IR_{kiij}$).
- B_{kij} – Gross margin that can be earned ha^{-1} for crop j, at stage i, on plot k. ($B_{kij} = MP_{kij} * YR_{kij} - C_{kij}$).

8.3.4 Variables

- L_k – Total area of land allocated for crop production for plot type k.
- X_{kij} – Area of land, in hectares, that can be feasibly allocated to crop j, at stage i, on plot k.

8.3.5 Objective Function

Maximize

$$f = \sum_{k=1}^{l} \sum_{i=1}^{N_k} \sum_{j=1}^{M_{ki}} X_{kij} B_{kij} \tag{8.1}$$

In equation 8.1, k represents the plot types. $k = 1$ indicates the single-crop plots, $k = 2$ indicates the double-crop plots, and so on. For each plot type k, i is indicative of the number of groups of crops that are grown in sequence throughout the year. For $k = 1$, N_k (or N_1) will be equivalent to 1. This will represent the group of crops that are grown all year round. For k = 2, $N_k = 2$. This will represent two groups of crops that are grown in sequence throughout the year. These are the summer and winter crop groups. The explanation is similar for $k = 3$, and so on. For each sequential crop group i, grown on plot k, j will represent the individual crops grown. For $k = 1$ and $i = 1$, j will be indicative of all the perennial crops grown. For $k = 2$ and $i = 1$, j will be indicative of all the summer crops grown. For $k = 2$ and $i = 2$, j will be indicative of all the winter crops grown, and so on.

Equation 8.1 is subject to the land and irrigated water allocation constraints given in sections 8.3.6 and 8.3.7 below. The gross benefits B_{kij} that can be earned per crop must also satisfy the non-negative constraint given in section 8.3.8 below.

8.3.6 Land Constraints

Feasible solutions must satisfy the lower and upper bound constraint of the plot type k. This constraint is given in equation 8.2 below.

$$Lb\_P_k \leq L_k \leq Ub\_P_k \quad \forall k, i, j \tag{8.2}$$

The sum of the hectares allocated for each plot type k must be less than or equal to T. This constraint is given by equation 8.3 below.

$$\sum_{k=1}^{l} L_k \leq T \tag{8.3}$$

The sum of the hectares allocated for each crop j, at stage i, on plot k, must be less than or equal to the total area of land allocated for crop production on plot type k. This constraint is given by equation 8.4 below.

$$\sum_{j}^{M_{ki}} X_{kij} \leq L_k \quad \forall k, i \tag{8.4}$$

The lower and upper bound constraint for each crop must be satisfied. This constraint is given by equation 8.5 below.

$$Lb_{kij} \leq X_{kij} \leq Ub_{kij} \quad \forall k, i, j \tag{8.5}$$

8.3.7 Irrigation Constraints

The total volume of irrigated water required for the production of all crops, within the year, must be less than or equal to the total volume of irrigated water that can be supplied to the given area of land. This constraint considers that some crops may require more irrigated water then what is supplied ha^{-1}. It is therefore the responsibility of the farmer to distribute his supply of irrigated water efficiently. This constraint is given by equation 8.6 below.

$$\sum_{k}\sum_{i}\sum_{j} IR_{kij} \leq TA \tag{8.6}$$

8.3.8 Non-negative Constraints

The gross profits that can be earned per crop must be greater than zero. This constraint is given by equation 8.7 below.

$$B_{kij} > 0 \quad \forall k, i, j \tag{8.7}$$

8.4 Case Study of the Taung Irrigation Scheme

The Taung Irrigation Scheme (TIS) is situated in the Taung District, in the North West Province of South Africa. It is a neighbouring irrigation scheme to the Vaalharts Irrigation Scheme (VIS). The VIS is one of the largest irrigation schemes in the world. TIS currently consists of a total of 3,764 ha of agricultural land (Smook *et al.*, 2008).

The irrigated water currently supplied to the TIS is drawn from the Vaal River, and is supplied via the Vaalharts Canal System. The Vaalharts Canal System also supplies irrigated water to the VIS. The irrigated water supplied to the TIS is supplied at a basic quota of 8,417 m^{3}ha^{-1}annum^{-1} to the farmers (Smook *et al.*, 2008).

Located close to the TIS is the Taung Dam. At full capacity, the dam consists of a total volume of 62.97 million m^3 of water. The dam was originally constructed to supply irrigated water to the TIS, but no infrastructure has been built to do so.

A survey (Smook *et al*., 2008) was done to determine if extending the existing TIS would be feasible and useful for developing new irrigated areas. If adjacent portions of land are seen to be feasible and useful for developing new irrigated areas, then the irrigated water supplied to the TIS will be drawn from the Taung Dam. The survey found that 3,315 ha are acceptable for agricultural production. It is also believed that agricultural production on this portion of land will match the high agricultural output of the neighbouring VIS.

The current expansion of the TIS will cater for 175 people that have been previously excluded from the land. A total of 1,750 ha (10 ha per person) will now be allocated to them for restitution. According to the choices of the local Department of Agriculture and the local farmers, the most suitable crops to be cultivated on this portion of land are those listed in Table 8.4.1 (Smook *et al*., 2008). The crops include perennial (p), summer (s) and winter (w) crops.

To determine solutions concerning the seasonal hectare allocations, amongst the various competing crops that are required to be grown, the Crop Water Requirement (CWR) and the average rainfall (AR) statistics need to be determined. The AR values are the average volumes of rain expected to fall during the developmental months of each crop. The CWR is obtained from Smook *et al*. (2008) and the average rainfall statistics are obtained from Maisela (2010).

The producer prices per ton (ZAR[3] t^{-1}) of yield are obtained from the Directorate Statistics and Economic Analysis (2012), and the Department of Agriculture, Forestry and Fisheries (2012). The yield expected ($t\ ha^{-1}$) per crop is determined from Agriculture & Environmental Affairs (2010). The water quota of 8,417 $m^3 ha^{-1} annum^{-1}$ remains the same. The cost of irrigated water is 8.77 cents/m^3 (Grove, 2008).

[3] ZAR stands for Zuid-Afrikaanse Rand. It is the Dutch translation of saying, "South African Rand." The Rand is the currency in South Africa.

Table 8.4.1: Taung irrigation scheme crop and average rainfall statistics

Crops	**CWR (mm)**	**AR (mm)**	**ZAR t^{-1}**	**t ha^{-1}**
Lucerne (p)	1,445	444.7	1,185.52	16.0
Tomato (s)	1,132	350.8	4,332.00	50.0
Pumpkin (s)	794	279.0	1,577.09	20.0
Maize (s)	979	279.0	1,321.25	9.0
Ground Nut (s)	912	339.5	5,076.00	3.0
Sunflower (s)	648	314.9	3,739.00	3.0
Barley (w)	530	58.3	2,083.27	6.0
Onion (w)	429	177.0	2,397.90	30.0
Potato (w)	365	152.8	2,463.00	28.0
Cabbage (w)	350	152.8	1,437.58	50.0

8.5 Testing and Evaluation

The non-heuristic specific parameters, required for the execution of the algorithms had been set according to the values given in Tables 8.4.2 and 8.4.3. The lower and upper bound settings for the different plot types are given in Table 8.4.2.

Table 8.4.2: Lower and upper bounds for each plot type

Plot Types	**Bounds (ha)**	
	$\boldsymbol{Lb\_P_k}$	$\boldsymbol{Ub\_P_k}$
Single-crop	10	1,700
Double-crop	50	1,740

Table 8.4.3 gives the lower and upper bound settings, the land coverage fraction values, the cost of irrigated water and the operational costs for each crop. The large differences in the lower and upper bound values were to investigate the ability of the metaheuristic algorithms in determining solutions within a larger solution space. $F_{kij} \in [0,1]$. C_IR_{kij} is the cost of the irrigated water per hectare per crop (ZAR ha^{-1}). O_{kij} is set to a third of the producer prices per ton of yield (ZAR ha^{-1}).

Table 8.4.3: Non-heuristic specific parameters required for the execution of the algorithms

Crops	Lb_{kij}	Ub_{kij}	F_{kij}	C_IR_{kij}	O_{kij}
Lucerne	10	1,700	1	877.26	6,259.52
Tomato	10	1,740	1	685.11	71,478.00
Pumpkin	10	1,740	1	451.66	10,408.80
Maize	10	1,740	1	613.90	3,924.09
Groundnut	10	1,740	1	502.08	5,025.24
Sunflower	10	1,740	1	292.13	3,701.61
Barley	12.5	1,740	1	413.68	4,124.88
Onion	12.5	1,740	1	221.00	23,739.30
Potato	12.5	1,740	1	186.10	22,758.12
Cabbage	12.5	1,740	1	172.94	23,720.00

The initial parameter settings for the LS metaheuristic algorithms were set as follows:

- BPA – The $listSize$ was set at 20. The $noOfIterations$ was set at 100,000. p_a was set at 0.2.
- IBPA – The $listSize$ was set at 20. The $noOfIterations$ was set at 5,000. The $stepsPerChange$ was set at 20.
- LADA – The $listSize$ was set at 20. The $noOfIterations$ was set at 50,000. m was set at 3.
- TS – The $tabuListSize$ was set at 7. The $candidateListSize$ was set at 20. The $noOfIterations$ was set at 5,000.
- SA – The $stepsPerChange$ was set at 100. T was set at 230. F was set at 0.01. α was set at 0.99.

The initial parameters for the population based metaheuristic algorithms were set as follows:

- CS – The number of host bird nests n was set at 20. The $noOfIterations$ was set at 100,000. p_a was set at 0.25.
- FA – The number of fireflies n was set at 20. The $noOfIterations$ was set at 5,000. α was set at 0.25, β_0 at 0.2 and γ at 1.
- GSO – The number of glow-worms n was set at 20. The $noOfIterations$ was set at 5,000. l_0 was set at 1, r_0 at 1.2, r_s at 1.5, ρ at 0.4, γ at 0.6, β at 0.08, st at 0.3 and N_d at 10.
- GA – The number of individuals n was set at 20. The $maxNoOfGenerations$ was set at 5,000. $cRate$ was set at 0.8. $mRate$ was set at 0.05 $(1/n)$.

The parameter settings are set very similar to the settings given in chapter seven, except for a few parameters. The changed parameter settings are; m in LADA, F in SA, the $noOfIterations$ parameters and the $maxNoOfGenerations$ parameter for GA. The $noOfIterations$ of BPA, IBPA, LADA, TS, CS, FA and GSO, the $maxNoOfGenerations$ of GA and the parameter settings of SA ensured that each algorithm executed for 100,000 objective function evaluations. Similar to chapter seven, each algorithm was run 100 times, using randomly generated population sets for each run. As explained in chapter seven, the population sets were used as input parameters for each of the algorithms, for each of the 100 runs.

From the 100 best solutions determined, by each metaheuristic algorithm, the results of the best and average solutions have been documented. Using the population of the 100 best solutions of each algorithm, the 95% confidence interval[4] (95% CI) values have also been calculated for the execution times and fitness values (total gross profits earned). The results are explained below.

Table 8.4.4 give the statistics of the average execution times (AVG) in milliseconds (ms), and the 95% CI values of each metaheuristic algorithm.

Table 8.4.4:The average execution times, in milliseconds, and the 95% CI values of each metaheuristic algorithm

Methods	**AVG (ms)**	**95% CI**
BPA	229	AVG $\pm$ 3
IBPA	223	AVG $\pm$ 3
LADA	147	AVG $\pm$ 2
TS	184	AVG $\pm$ 5
SA	212	AVG $\pm$ 3
CS	884	AVG $\pm$ 2
FA	3,455	AVG $\pm$ 6
GSO	751	AVG $\pm$ 3
GA	915	AVG $\pm$ 3

[4] In statistics a Confidence Interval (CI) indicates the reliability of an interval estimate of population parameters. 95% CI means to be 95% certain that the population parameters will lie within the interval estimate range.

For this case study, one can again observe that LADA executed the fastest overall. The average execution times of the other LS algorithms were all comparable. The execution times of the population based algorithms are again much slower than that of the LS algorithms. Once more, FA took the longest time to execute overall. The average execution times of CS, GSO and GA were also relatively comparable. The nested *for* loop in FA is the reason for it being computationally expensive.

In observing the 95% CI values we conclude that the execution times of the algorithms had again been fairly consistent. A visual representation of the statistical values given in Table 8.4.4 is shown in Figure 8.4.1 below. The 95% CI values are represented by the black interval estimates.

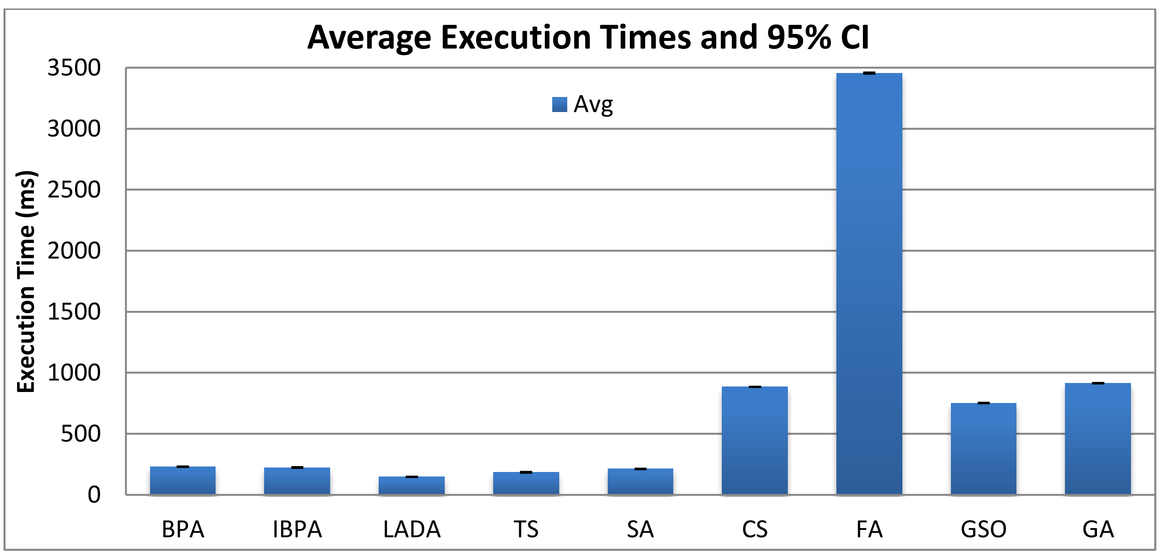

Figure 8.4.1: The average execution times, in milliseconds (ms), and the 95% CI values of each metaheuristic algorithm

Table 8.4.5 gives the statistical values of the overall best fitness values (BFV) and average best fitness values (ABFV) for each metaheuristic algorithm. The fitness values are the total gross profits earned. The 95% CI values for the fitness value populations of each algorithm is also given.

Table 8.4.5: Statistics for the best fitness values (BFV), average best fitness values (ABFV) and 95% CI values

Methods	BFV (ZAR)	ABFV (ZAR)	95% CI
BPA	295,382,093	287,575,514	ABFV $\pm$ 732,543
IBPA	296,166,629	288,864,091	ABFV $\pm$ 756,861
LADA	296,241,511	280,062,612	ABFV $\pm$ 1,352,737
TS	298,765,873	296,886,105	ABFV $\pm$ 185,479
SA	294,824,404	288,363,133	ABFV $\pm$ 866,622
CS	290,770,383	282,000,392	ABFV $\pm$ 936,537
FA	297,967,538	295,623,620	ABFV $\pm$ 195,076
GSO	299,551,069	280,488,876	ABFV $\pm$ 6,352,385
GA	286,477,093	264,550,148	ABFV $\pm$ 1,502,171

From Table 8.4.5, it is observed that GSO determined the highest BFV. This was followed by TS, FA, LADA, IBPA, BPA, SA, CS and then GA. On average, TS performed the best. This was followed by FA, IBPA, SA, BPA, CS, GSO, LADA and then GA. For the LS algorithms, although LADAs' BFV was higher than IBPA, BPA and SA, its average performance was the worst overall. This proves that LADA had the ability to determine good solutions, although it performed relatively poorly on average. From all metaheuristic solution, GA performed the worst overall.

A graphical comparison of the algorithms best and average fitness values, as determined from Table 8.4.5, is shown in Figure 8.4.2. The 95% CI values are represented by the black interval estimates over the average fitness values.

The solutions found by the algorithms were in a solution space of constantly changing plot type hectare allocations. The hectare allocations of each plot type needed to be determined first before the hectare allocations of the crops. The hectare allocations had to satisfy the land constraints given in section 8.3.6.

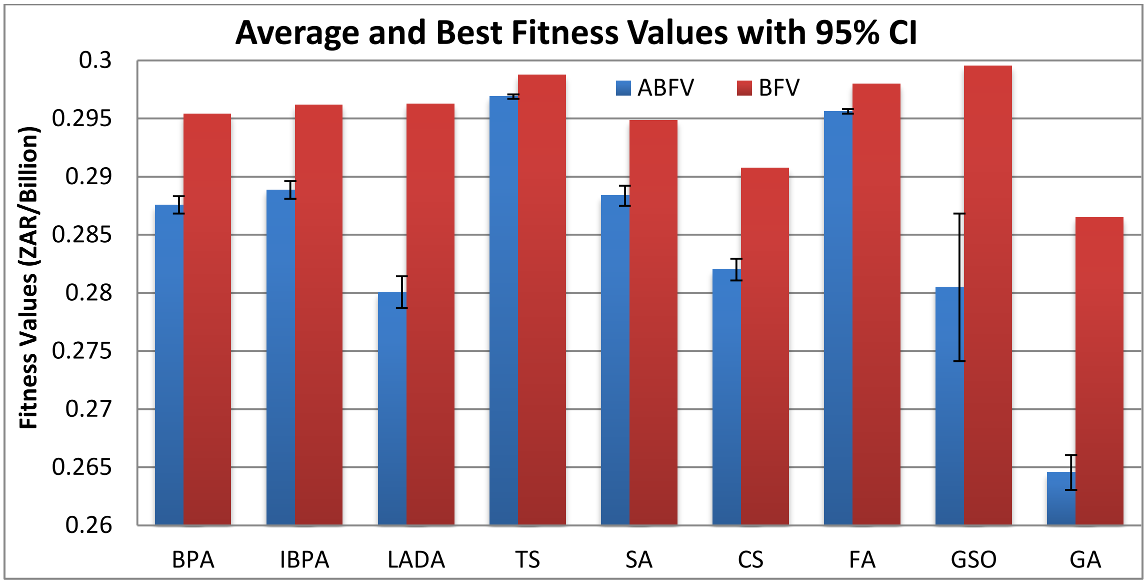

Figure 8.4.2: A comparison of each algorithm's best and average fitness values determined, along with the 95% CI estimates

For each algorithm, the best solution determined from the population of solutions at iteration t, for plot type hectare allocations p, will not necessarily be the best solution at iteration $(t+1)$ for plot type hectare allocations $(p+1)$. The change in the plot type hectare allocations at iteration $(t+1)$ will change the crop hectare allocations accordingly, so the land constraints do not break. The constantly changing dimensions of the solution space make it very difficult for the algorithms to perform exploitation. This makes it difficult to determine effective solutions.

Under the circumstance of the constantly changing dimensions of the solution space, TS and FA had performed the most consistently. This is confirmed by their low 95% CI fitness values. BPA had the third lowest 95% CI fitness value. This is followed by IBPA, SA, CS, LADA, GA and then GSO. By observing and comparing each algorithm's BFV, ABFV and 95% CI fitness value solutions one can conclude that TS had been the strongest metaheuristic algorithm, in determining solutions for this particular optimization problem. This is followed by FA, IBPA, SA, BPA, GSO, LADA, CS and then GA.

Although GSOs' average performance is worse than CS, its best fitness value and its high 95% CI fitness value prove that though it determined many good solutions, it also had poor solutions leading to a lower average.

The strength of TS, in performing the best overall, is due to its strong exploitation ability. At iteration t, generating a candidate list of solutions allows TS to maximize its exploitation within the local neighbourhood structure of the solution space, for plot type hectare allocations p. The best candidate solution determined at iteration t will be the best solution found for plot type hectare allocations p, but as explained earlier, it will not necessarily be the best "working" solution at iteration $(t + 1)$, for plot type hectare allocations $(p + 1)$. However, if $(p + 1)$ is very similar to p, then the working solution at iteration $(t + 1)$ will become very valuable in trying to effectively exploit the local neighbourhood structure of the solution space even further. The possibility of $(p + 1)$ being similar to p, and in using the best candidate solution from iteration t as the working solution at iteration $(t + 1)$, has further encouraged exploitation. This is the reason why TS had performed well.

Similar to TS, IBPA uses a "current" solution to perform exploitation at each iteration t, for a certain number of "steps per change". The solution chosen as the current solution at iteration t is restricted to the solutions listed on the Performance List (PL). Any "working" solution generated from the current solution, at iteration t, will therefore not necessarily be related to the current solution chosen at iteration $(t + 1)$. This statement holds even if any working solution generated updates the PL. The possibility of further exploiting a local neighbourhood structure of the solution space if, p is very similar to $(p + 1)$ is therefore minimized.

The purpose of maintaining updated lists of their best solutions found, for BPA, IBPA and LADA, is to facilitate exploration of the solution space. Performing local search facilitates exploitation. For this particular optimization problem, IBPA and BPA show a better balance in performing exploration and exploitation, compared to LADA. This is in comparism with SA in terms of their performances. SA has a naturally good balance in its ability to perform exploration and exploitation. LADA seems to be stronger in its explorative ability. This explains its relatively high BFV solution and its relatively low ABFV performance.

For the population based algorithms, the strength of FA and GSO, in determining the best fitness solutions, is attributed to the algorithms versatility in being able to accept both improved and worse solutions with each iteration. In FA, as the fireflies get attracted towards brighter fireflies, at iteration t, some will accept improved solutions while others will accept worse solutions within the local neighbourhood structures of the solution space. The solutions found, that are classified as being either improved or worse, depend entirely on the plot type hectare allocations p, at iteration t. However, at iteration $(t+1)$, the sorting of the fireflies will take place according to the plot type hectare allocations $(p+1)$ and not p. Therefore, what appears to be improved solutions at iteration t, for p, might not necessarily be an improved solution at iteration $(t+1)$ for $(p+1)$. Similarly, what appears to be a worse solution at iteration t, for p, might not necessarily be a worse solution at iteration $(t+1)$ for $(p+1)$. The versatility of FA, in accepting both improved and worse solutions, has shown to be very valuable for this particular optimization problem, for a population based algorithm.

In GSO, a glow-worm will accept an improved or worse solution in moving towards another glow-worm with a higher level of luciferin than itself. Similar to FA, this ability is shown to be very valuable for this particular optimization problem. GSOs' ABFV is however relatively low, compared to FA and CS. Interestingly enough, it also has the highest 95% CI fitness value. The reason for the instability of its performances is due to its ability to deliberately cause group-like separations of the glow-worms throughout the neighbourhood structures of the solution space. The separations are achieved by reducing the glow-worms vision ranges as the number of iterations increase, and in limiting the maximum number of neighbours that a glow-worm is allowed to have. The group-like separations result in fewer glow-worms searching the local neighbourhood structures of the solution space. This technique is strong in exploration but lacks in exploitation. Stronger explorative abilities have shown to be more beneficial for the population based algorithms, for this particular type of optimization problem. This is due to the constantly changing dimensions of the solution space. However, the weakness in GSOs' exploitative ability reduces the probability of it performing consistently on average. This explains its relatively low ABFV performance.

For each host bird's nest solution in the population, CS only accepts new nest solutions if it improves on the host bird's nest solutions in the population. The new nest solutions are generated by using the best nest solution from the previous iteration, in performing levy flights. However, as explained earlier, what appears to be the best nest solution at iteration $(t-1)$, for plot type hectare allocations $(p-1)$, will not necessarily be the best nest solution to be used at iteration t, using plot type hectare allocations p. Therefore, due to the constant changes in the dimensions of the solution space, performing levy flights will not result in the most effective exploitation. The probability of the host bird discovering intrusions facilitates exploration. This has given CS the best chance at determining improved solutions.

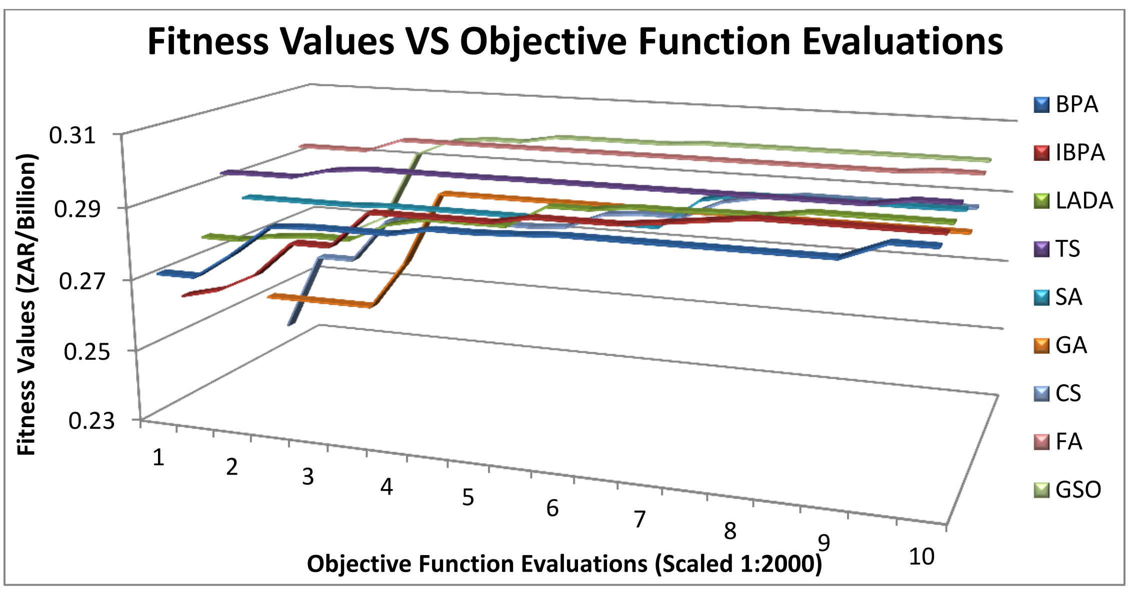

Figure 8.4.3: The performance of the metaheuristic algorithms in determining their overall best fitness value solutions

Figure 8.4.3 shows the performances of the metaheuristic algorithms in determining their BFV solutions. For the LS metaheuristic algorithms, it can be seen that TS has clearly outperformed the other algorithms, in determining its BFV. SA had initially progressed at a very fast rate, up to about 10,000 objective function evaluations, compared to BPA, IBPA and LADA. BPA, IBPA and LADA performed similarly in progressively improving on their BFV performances. TS found its best fitness value at around 90,000 objective function evaluations.

For the population based algorithms, FA found improved solutions at the fastest rate up to around 25,000 objective function evaluations. At this point, GSO determined a solution similar to FA. At around 63,000 objective function evaluations, GSO had determined the best fitness value from all metaheuristic algorithms. FA also found its best fitness value at around 90,000 objective function evaluations. CS showed steady increases in determining improved solutions. At around 70,000 objective function evaluations, CS found a neighbourhood within the solution space which had a solution that was better than GAs' best solution. GA found its best solution at around 34,000 objective function evaluations.

Table 8.4.6: Statistics of the irrigated water requirements (IWR) and variable costs of production (VCP) for the best solutions found

Methods	**IWR (m^3)**	**VCP (ZAR)**
BPA	16,922,183	147,701,718
IBPA	16,961,536	148,093,316
LADA	17,244,651	74,544,333
TS	17,142,919	149,397,333
SA	17,070,610	147,446,530
CS	16,971,534	145,436,812
FA	16,962,160	148,980,411
GSO	17,052,921	149,772,256
GA	17,103,618	143,339,455

Table 8.4.6 gives the statistics of the IWR and the VCP values for the best solution determined by each metaheuristic algorithm. BPAs' solution required the least volume of irrigated water. This was followed by IBPA, FA, CS, GSO, SA, GA, TS and then LADA.

At a cost of ZAR 0.0877 m^{-3}, the cost of BPAs' irrigated water is ZAR 1,484,075. The IWR of IBPA, FA, CS, GSO, SA, GA, TS and LADA was a volume of 39,353 m^3, 39,977 m^3, 49,351 m^3, 130,738 m^3, 148,427 m^3, 181,435 m^3, 220,736 m^3 and 322,468 m^3 more than BPAs' IWR respectively. At a water quota of 8,417 $m^3ha^{-1}annum^{-1}$, BPAs' IWR value would have supplied irrigated water to 4, 5, 6, 16, 17, 22, 26 and 38 ha's less than the IWR of IBPA, FA, CS, GSO, SA, GA, TS and LADA respectively.

From Table 8.4.6, it is also observed that the VCP values of BPA, IBPA, TS, SA, CS, FA, GSO and GA are similar. Interestingly enough, LADAs' VCP value is about half of the VCP values of each of the other heuristic algorithms. From all heuristic algorithms, except LADA, GSO has the highest VCP values and GA has the lowest VCP values. Compared to GSO, LADAs' VCP value is ZAR 75,227,923 less. In comparison to GA, LADAs' VCP value is ZAR 68,795,122 less. Although, GSO determined a best overall solution that earned an extra gross profit of ZAR 3,309,558, and required a volume of 191,730 m^3 less of irrigated water in comparison to LADAs' best solution, the remarkable saving in LADAs' VCP value means that LADA determined the most economically feasible solution compared to the other metaheuristic algorithms. A graphical representation of the IWR's, as determined from Table 8.4.6, is shown in Figure 8.4.4.

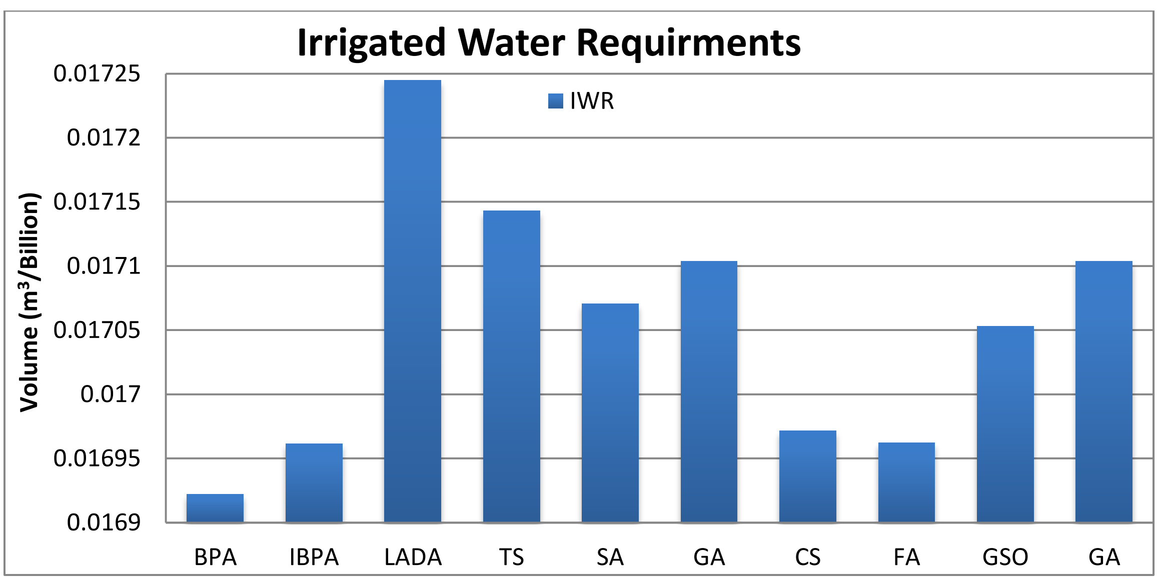

Figure 8.4.4: Irrigated water requirements (IWR) of the best metaheuristic solutions

Table 8.4.7 gives the plot type hectare allocations for the best solution found by each metaheuristic algorithm. Except for LADA, each metaheuristic algorithm determined that the total gross profits will be greater in allocating more land for the double-crop plots. LADAs' best solution determined that allocating more land to the single-crop plots would be better. This is regardless of lucernes' relatively high IWR and relatively low producer price t^{-1} value, compared to all other crops.

Table 8.4.7: Plot type hectare allocations for each metaheuristic algorithm

Methods	Single-Crop Plots	Double-Crop Plots
BPA	17	1,733
IBPA	12	1,738
LADA	956	794
TS	14	1,736
SA	18	1,732
CS	16	1,734
FA	13	1,737
GSO	14	1,736
GA	13	1,737

Figure 8.4.5 gives a graphical comparison of the seasonal hectare allocations for each crop, for the best solution determined by each metaheuristic algorithm. For the single-crop plots of land, all algorithms, except LADA, determined similar hectare allocations for lucerne. LADAs' hectare allocation was clearly higher. For the double-crop plots of land, all metaheuristic algorithms allocated the most area of land to tomato, onion and cabbage. The large hectare allocation for tomato is due to its high yield ha^{-1} and high producer price t^{-1} value. Similar hectare allocations were determined for pumpkin, maize, ground nuts and sunflower by each algorithm. GAs' relatively higher hectare allocation for barley contributed to the relatively poor best solution obtained.

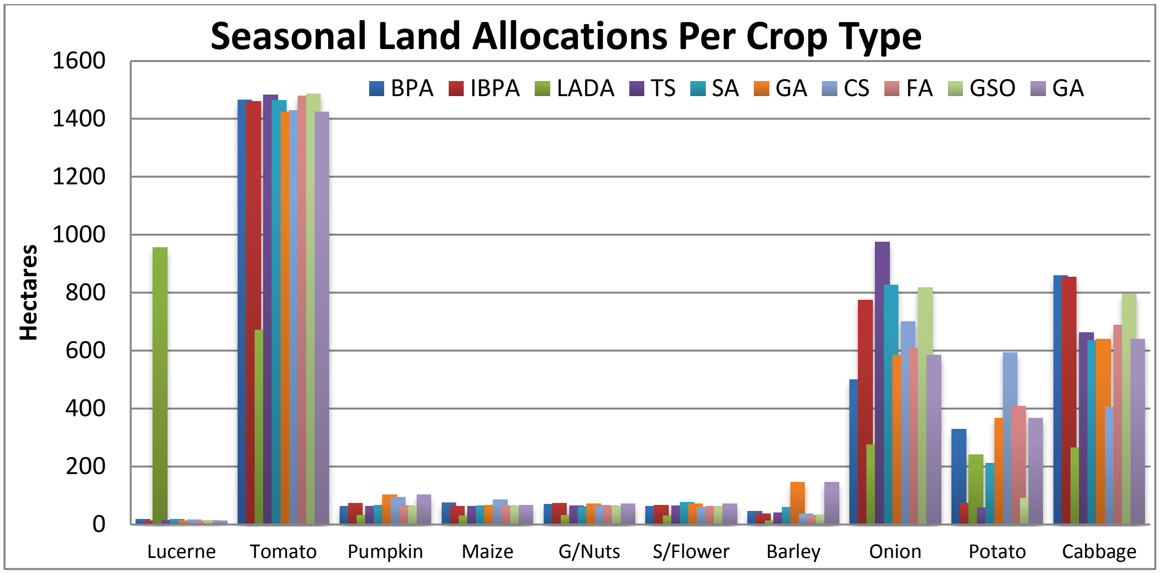

Figure 8.4.5: A comparison of the hectare allocations, per crop, for the best solution found by each metaheuristic algorithm

Tables 8.4.8 and 8.4.9 give the statistical values of each crops hectare allocations (ha's crop^{-1}), IWR and VCP for the best solution determined by each metaheuristic algorithm.

The program was written with the Java programming language. It was programmed using the Netbeans® 7.0 Integrated Development Environment. All simulations were run on the same platform. The computer used had a Windows® 7 Enterprise operating system, an Intel® Celeron® Processor 430, 3 GB of RAM and a 500GB hard-drive.

In developing object oriented versions of these metaheuristic algorithms, the LS algorithms were relatively easier to implement. The LS algorithms also require minimal parameter settings. The SI algorithms were relatively harder to implement. However, compared to GSO, FA and CS were relatively easier to implement. For the SI algorithms, CS requires the least number of parameter settings.

Table 8.4.8: Crop statistics of the best solution determined by each metaheuristic algorithm

Crops	Methods	ha's crop^{-1}	IWR (m^3)	VCP (ZAR)
Lucerne	BPA	17	169,016	120,587
	IBPA	12	123,883	88,386
	LADA	956	9,560,965	6,821,407
	TS	14	138,942	99,130
	SA	18	175,621	125,299
	CS	16	159,049	113,476
	FA	13	129,349	92,286
	GSO	14	145,019	103,466
	GA	13	128,561	91,724
Tomato	BPA	1,465	11,442,080	105,695,864
	IBPA	1,461	11,416,473	105,459,327
	LADA	671	5,242,001	48,422,824
	TS	1,483	11,587,560	107,039,732
	SA	1,463	11,426,259	105,549,719
	CS	1,429	11,161,035	103,099,717
	FA	1,479	11,553,491	106,725,022
	GSO	1,487	11,617,618	107,317,394
	GA	1,424	11,125,493	102,771,401
Pumpkin	BPA	62	318,126	670,872
	IBPA	73	375,268	791,375
	LADA	31	159,882	337,164
	TS	62	319,971	674,763
	SA	67	343,852	725,124
	CS	93	476,715	1,005,310
	FA	65	336,130	708,840
	GSO	62	320,703	676,306
	GA	103	532,815	1,123,615
Maize	BPA	75	522,969	339,033
	IBPA	63	443,563	287,555
	LADA	30	211,339	137,008
	TS	63	437,786	283,810
	SA	65	454,319	294,528
	CS	86	603,190	391,039
	FA	65	453,873	294,239
	GSO	62	434,061	281,395
	GA	67	468,012	303,405
Ground Nuts	BPA	69	392,341	378,794
	IBPA	73	416,717	402,328
	LADA	32	184,256	177,894
	TS	64	363,636	351,080
	SA	61	351,911	339,759
	CS	66	378,980	365,895
	FA	65	373,218	360,331
	GSO	62	355,404	343,133
	GA	72	410,972	396,781

Table 8.4.9: Crop statistics of the best solution determined by each metaheuristic algorithm

Crops	Methods	ha's crop^{-1}	IWR (m^3)	VCP (ZAR)
Sunflower	BPA	63	211,220	253,245
	IBPA	67	223,812	268,341
	LADA	30	99,098	118,815
	TS	65	215,246	258,072
	SA	77	255,319	306,118
	CS	60	201,401	241,472
	FA	63	209,286	250,925
	GSO	62	206,495	247,579
	GA	71	236,134	283,116
Barley	BPA	46	216,110	207,935
	IBPA	36	171,475	164,988
	LADA	12	57,471	55,297
	TS	41	195,287	187,899
	SA	59	278,448	267,915
	CS	36	170,083	163,649
	FA	31	148,232	142,625
	GSO	33	154,230	148,395
	GA	146	689,062	662,995
Onion	BPA	499	1,258,048	11,961,592
	IBPA	775	1,952,043	18,560,133
	LADA	276	695,752	6,615,253
	TS	974	2,455,286	23,345,004
	SA	827	2,084,191	19,816,604
	CS	700	1,764,290	16,774,969
	FA	609	1,534,750	14,592,494
	GSO	817	2,059,855	19,585,220
	GA	584	1,471,225	13,988,493
Potato	BPA	329	699,027	7,558,257
	IBPA	73	155,092	1,676,941
	LADA	241	512,445	5,540,829
	TS	57	121,629	1,315,123
	SA	211	448,211	4,846,302
	CS	593	1,257,325	13,594,878
	FA	409	867,245	9,377,127
	GSO	90	191,282	2,068,249
	GA	367	778,789	8,420,687
Cabbage	BPA	859	1,693,246	20,515,539
	IBPA	854	1,683,210	20,393,942
	LADA	264	521,442	6,317,842
	TS	663	1,307,576	15,842,720
	SA	635	1,252,479	15,175,162
	CS	405	799,466	9,686,407
	FA	688	1,356,586	16,436,522
	GSO	795	1,568,254	19,001,119
	GA	640	1,262,555	15,297,238

8.6 Conclusion

Increase in crop production costs, shortages in food supply, and increase in population growth have made the need for optimized solutions in crop planning mandatory. However, determining optimized solutions is not enough. In trying to meet the growing demand for food in the future, it is important that new irrigation schemes be developed to increase agricultural output.

The planning of new irrigation schemes require that optimized solutions be found for the seasonal hectare allocations of the crops to be grown within the year. The solutions found must seek to maximize the total gross profits that can be earned, in making the most efficient use of the limited resources available for crop production.

This chapter introduces an ACP mathematical model for a new irrigation scheme. The Taung Irrigation Scheme (TIS), situated in the North West Province of South Africa was used as the case study. The irrigation scheme is currently being expanded to cater for an extra 1,750 hectares of irrigated land. This portion of land is required to grow ten different types of crops. To determine solutions for this ACP problem, three new LS (BPA, IBPA and LADA) and three relatively new SI metaheuristic algorithms (GSO, CS and FA) have been investigated. Results of these are compared with the solutions of TS, SA and GA.

To ensure fairness in the performances of the metaheuristic algorithms, the algorithm specific parameter settings of TS, CS, FA and GSO were set according to recommended settings. Other parameter settings, such as the 'list' sizes, the 'population' sizes and the initial population sets were also set to be the same. The parameter settings ensured that the total number of objective function evaluations, per run, would be the same for each algorithm. Each metaheuristic algorithm was run 100 times. From these 100 runs, the overall best and average solutions for each algorithm have been documented.

The solutions found by the metaheuristic algorithms were in a solution space of constantly changing dimensions. This made it very difficult for the algorithms to determine effective solutions. The results show that GSO determined the best solution overall. On average, TS performed the best. Under this circumstance of constantly changing dimensions of the solution space, TS and FA had performed the most consistently. This is confirmed by their low 95% CI fitness values. By observing and comparing each algorithm's BFV, ABFV and 95% CI fitness value solutions, it is concluded that TS has been the strongest metaheuristic algorithm in determining solutions to this particular optimization problem. From all metaheuristic algorithms, however, LADA determined the most economically feasible solution.

An added advantage of LADA is its low execution time. For all metaheuristic algorithms, FA took the longest time to execute. For the population based algorithms, GSO had the fastest execution time. Although, GSOs' average performance was relatively low, its best solution and its high 95% CI fitness value proved that it had determined very good solutions. From all metaheuristic algorithms, GA performed the worst overall.

CHAPTER NINE
CONCLUSION AND FUTURE RESEARCH

9.1 Conclusion

This research introduces the Annual Crop Planning (ACP) problem for new and existing irrigation schemes. Due to increased costs, and the limited resources available for crop production, it has become very important that optimized solutions be found in determining resource allocation solutions in crop planning. Determining optimized solutions in making resource allocation decisions, at the level of an irrigation scheme, amongst the various competing crops that are to be produced within a production year is referred to as an ACP problem. The objective of determining solutions to an ACP problem is to maximize the total gross profits that can be earned in making resource allocation decisions. The resources that are required to be optimized include the limited area of agricultural land, the irrigated water supply and the variable costs associated with crop production. In determining solutions, it should be considered that crops differ in their plant requirements. Different types of crops also grow for a different number of days, and have different planting and harvesting schedules. Other types of factors that must be considered in determining solutions include the crop yields, the climatic conditions, the market demand conditions and the fluctuating markets costs, amongst others. These factors will affect the resource allocations for each crop and the total gross profits earned at the end of a production year.

ACP is an NP-Hard type optimization problem, which is formulated as a multiple knapsack problem. Due to the complexity involved in determining solutions, and the uncertainty of several factors, it is not advisable that exact algorithms be used to determine solutions. Exact methods guarantee that the optimal solution will be found, however, for NP-Hard optimization problems there is no guarantee that an optimal solution can be found within reasonable computational time. For NP-Hard type optimization problems, heuristic algorithms are preferred. Heuristic algorithms determine near-optimal solutions within polynomial time (P). Near-optimal solutions are acceptable due to the reduction gained in the computational time involved with determining feasible solutions.

To determine near-optimal solutions to the ACP problems presented in this dissertation, three new Local Search (LS) metaheuristic algorithms have been introduced. The new LS algorithms are called the Best Performance Algorithm (BPA), the Iterative Best Performance Algorithm (IBPA) and the Largest Absolute Difference Algorithm (LADA). The motivation in developing these algorithms was to investigate techniques that could be used to determine effective solutions to difficult optimization problems at low computational costs. Another reason was to make a contribution to the field of optimization. The new algorithms developed are based on techniques that maintain updated lists of their best solutions found. To determine the relative merits of the solutions found by these new LS algorithms, their solutions have been compared with the solutions of two other well-known LS metaheuristic algorithms. These algorithms are Tabu Search (TS) and Simulated Annealing (SA).

This research also investigates the abilities of three recently developed Swarm Intelligence (SI) metaheuristic algorithms, in determining solutions to the same ACP problems. These algorithms include Cuckoo Search (CS), the Firefly Algorithm (FA) and Glowworm Swarm Optimization (GSO). To the best of the authors' knowledge, no other research has been found that compares the performances of these particular SI algorithms in determining solutions to a crop planning problem. To determine the relative merits of the solutions found by these SI algorithms, their solutions have been compared against the solutions of another popular population based metaheuristic algorithm. This algorithm is the Genetic Algorithm (GA).

The performances of all metaheuristic algorithms have also been compared. The algorithms were compared based on their abilities to determine solutions to the ACP problems for a new and existing irrigation scheme. Comparisons of the algorithms' execution times were also done. In making comparisons, conclusions were drawn concerning the possible strengths and weaknesses of the three new LS and three relatively new SI metaheuristic algorithms, in their determination of solutions.

This research has also introduced two new ACP mathematical models. The mathematical models are intended to be used to determine resource allocation solutions to the ACP problems at both new and existing irrigation schemes. The ACP mathematical models have been formulated as instances of the Space Allocation Problem (SAP). Space allocation, in optimization, involves allocating a limited area of available space amongst the demanding entities that require space utilization (Silva, 2003). The limited space needs to be allocated in a way that gives the most amount of satisfaction to all demanding entities involved, in optimizing the problems' objective.

At existing irrigation schemes, the farm plot sizes for the single-crop plots, double-crop plots, triple-crops plots, etc., are usually fixed. The single-crop plots are used to grow perennial crops. Perennials include tree bearing crops, which are usually harvest once a year. Other types of perennial crops include those that are harvested several times within a year. An example of this crop is lucerne. Perennial crops grow all year around. The double-crop plots are used to grow two groups of crops that are grown in sequence within the year. These groups can include seasonal crops such as the summer and winter crops. Triple-crop plots are used to grow three groups of sequential crops within the year, and so on.

ACP for an existing irrigation scheme involves determining the seasonal hectare allocations amongst the various competing crops that are required to be grown on the single-crop, double-crop and triple-crop plots, etc. This is subject to the limited area of land that is available for crop production on these farm plots. An optimized solution allocates the limited resources amongst the various competing crops that are to be grown within a production year. The objective of making resource allocation decisions is to maximize the total gross profits that can be earned within a production year.

At a new irrigation scheme, the hectare allocations for the various competing crops and the hectare allocations of the plots types need to be determined. Determining the hectare allocations of the plot types is important. The plot type hectare allocations will generally become fixed once decided upon. The hectare allocations of the plot types and the crops to be selected are influenced by the geographical location of the irrigation scheme. At a specific geographical location, several factors will need to be considered in determining solutions. These factors include the climatic conditions, the adaptability of the crops for sustainable crop production, the crop yields, the forecasted producer prices, the various costs associated with crop production and the market demand conditions, amongst others. The aim of determining solutions to this ACP is also to optimize the resource allocations amongst the various competing crops that are to be grown. The objective is also to maximize the total gross profits earned. The profits earned must contribute towards the financial investment involved with the development of the irrigation scheme. Once the resource allocation decisions are made for the first year, the ACP model for an existing irrigation scheme can then be used to determine solutions for the following years.

To ensure fairness in the execution of the metaheuristic algorithms, many parameter settings were set according to recommended settings found in literature. Many other parameter settings were also set to be the same. The parameter settings for each of the algorithms, for each problem instance, ensured that the total number of objective function evaluations would be the same at each run. For each problem instance, each metaheuristic algorithm was run 100 times. The 100 runs were to determine the overall best and average solutions determined. From the solutions determined, several comparisons were made in the algorithm’s abilities to determine solutions. Comparisons of the average execution times, the algorithm’s performances, the irrigated water allocations, the variable costs associated with crop production and the hectare allocations to the various competing crops were made for each problem instance. For the case study of an existing irrigation scheme, the solutions determined were compared with the statistics of the current agricultural practices at the scheme.

The metaheuristic solutions for the existing irrigation scheme showed that, each algorithm determined superior solutions to that of the current agricultural practices. Each algorithm's overall best solution determined seasonal hectare allocations that showed increased gross profits and reduced volumes of irrigated water allocations. Each algorithm determined that primarily increasing the hectare allocations for cotton and ground nuts, and decreasing the hectare allocations for maize were the main differences in determining improved solutions. From all metaheuristic algorithms, CS determined the best solutions and was the most consistent on average. It was concluded that CS was the best metaheuristic algorithm for this particular optimization problem. From all LS algorithms, BPA and IBPA performed the best. The algorithm that performed the worst overall was GA. LADA had the fastest average execution time. FAs' average execution time was the worst overall.

Determining solutions for a new irrigation scheme is more difficult than determining solutions for an existing irrigation scheme. The solutions found by the metaheuristic algorithms for this ACP problem were in a solution space of constantly changing dimensions. This made it increasingly difficult for the algorithms to determine effective solutions. Under this circumstance, the results show that GSO determined the best solution overall. On average, TS performed the best. The most consistent metaheuristic performances were given by TS and FA. Although GSO determined the best solution overall, by observing and comparing each algorithm's best overall solution, average best solution and 95% confidence interval fitness values, it is concluded that TS was the strongest metaheuristic algorithm for this particular optimization problem. From all metaheuristic algorithms, however, LADA determined the most economically feasible solution. The required financial investment determined by LADAs' best solution was about half of the financial investments required by each of the other metaheuristic algorithm's best solution. The gross profit earned for LADAs' best solution was also only marginally inferior to GSOs' best solution. The average execution time for LADA was again the fastest overall. FAs' execution time was again the worst overall.

In general, it has been observed that the average execution times for the LS metaheuristic algorithms are much faster than that of the population based algorithms. The solutions of the

LS algorithms were also very competitive. LS algorithms are much easier to implement, and require minimal parameter settings. For the SI algorithms, FA and CS were relatively easier to implement compared to GSO. From these algorithms, CS requires the least number of parameter settings. However, GSO executes the fastest.

The performance of the three new LS metaheuristic algorithms, in determining solutions to both ACP problems is shown to be very competitive. The techniques used to maintain updated lists of the best solutions found, have proven to be very effective in determining solutions at low computational costs. BPA and IBPA show good balances in exploring and exploiting the local neighborhood structures of the solutions space. LADA has a stronger explorative ability.

9.2 Future Research

There are further opportunities to improve both the model and solutions to the ACP problem, especially based on several other case studies that exist. This will enhance the development of a more robust model that will be a replication of the generic case of the ACP model. Furthermore, more experiments might still be required to test the robustness and efficiency of the three new LS metaheuristic algorithms introduced in this research. Specifically, further research can be done on the LADA algorithm whose main weakness lies in its exploitative ability which might not be well suited for problems that require strong exploitation. Further techniques can be developed or hybridizations made to improve on LADAs' exploitative ability. Possible hybridizations with LADA include hybridizations with GA and/or TS. Possible hybridizations of BPA and IBPA with other algorithms should also be investigated.

In ACP, further research can be done in determining solutions to inter-cropping practices. To encourage further research in determining solutions to the ACP problems for new and existing irrigation schemes, a collection of 12 test benchmark datasets have been compiled and is included in Appendix A. These datasets have been used to test the performances of the algorithms in determining solutions to larger instances of the ACP problems.

REFERENCES

Acquaah, G. (2004). "Principles of crop production: theory, techniques, and technology," 2nd Edition, Prentice Hall.

Adamu M.O. and Adewumi A.O. (2012). "Metaheuristics for scheduling on parallel machine to minimize weighted number of early and tardy jobs". Int J. of Physical Sciences Vol. 7(10), pp. 1641 - 1652, 2 March.

Adewumi, A.O (2010). Some improved genetic-algorithms based heuristics for global optimization with innovative applications. PhD. Thesis. Faculty of Science, University of the Witwatersrand, Johannesburg, South Africa.

Adewumi, A.O. and Ali, M. M. (2010). "A multi-level genetic algorithm for a multi-stage space allocation problem," Mathematical and Computer Modeling, Vol. 51(1-2), pp. 109-126.

Adeyemo, J. and Otieno, F. (2010a). "Maximum irrigation benefit using multi-objective differential evolution algorithm (MDEA)," Int. J. of Sustainable Development, Vol. 1(2), pp. 39-44.

Adeyemo, J. Bux, F. and Otieno, F. (2010b). "Differential evolution algorithm for crop planning: single and multi-objective optimization model," Int. J. of the Physical Sciences, Vol. 5(10), pp. 1592-1599.

Agriculture & Environmental Affairs (2010). "Expected Yields," [Online] vailable: www.kzndae.gov.za.

Allen, R. G. Pereira, L. S. Raes, D. and Smith, M. (1998). "Crop Evapotranspiration (guidelines for computing crop water requirements)," FAO Irrigation and Drainage Paper No. 56, [Online] Available: www.fao.org.

Ashraf, M. and Majeed, A. (2006). "Water requirements of major crops for different agro-climatic zones of Balochistan," The World Conservation Union (IUCN) Pakistan, Water Programme.

Astera, M. (2010). "The Ideal Soil: A Handbook for the New Agriculture," 1st edition, [Online]Available: www.soilminerals.com.

Bai, R. (2005). “An investigation of novel approaches for optimising retail shelf space allocation,” Ph.D. thesis, University of Nottingham.

Blaylock, A. D. (1994). “Soil salinity, salt tolerance, and growth potential of horticultural and landscape plants,” College of Agriculture, Univ. of Wyoming.

Blum, C. and Merkle, D. (2008). “Swarm Intelligence Introduction and Applications,” Springer Verlag Berlin Heidelberg.

Blum, C and Roli, A. (2003). “Metaheuristics in Combinatorial Optimization: Overview and Conceptual Comparison,” ACM Computing Surveys, Vol. 35(3).

Boyd, S and Vandenberghe, L. (2004). “Convex Optimization,” Cambridge University Press, UK.

Brouwer, C. and Heibloem, M. (1986). “Irrigation Water Management: Irrigation Water Needs,” FAO Land and Water Development Division, [Online] Available: www.fao.org.

Brouwer, C. Prins, K. Kay, M. and Heibloem, M. (1990). “Irrigation Water Management: Irrigation Methods,” FAO Land and Water Development Division, [Online] Available: www.fao.org.

Chandy, K.T. (1993). “Water Requirement of Different Crops,” Agricultural & Environmental Education, Booklet No. 226, Irrigation: IRS – 11. [Online] Available: www.inseda.org

Charles A.F. (1986). “Multiple cropping systems,” Technology and Engineering, Macmillan Pub. Co.

Department of Agriculture, Forestry and Fisheries (2012). “Trends in the agricultural sector 2012”, [Online] Available: www.daff.gov.za.

Directorate Statistics and Economic Analysis (2012). “Abstract of agricultural statistics 2012,” Department of Agriculture, Forestry and Fisheries, [Online] Available: www.nda.agric.za.

Domshlak, C. Prestwich, S. Rossi, F. Venable, K. B. and Walsh, T. (2006). “Hard and soft constraints for reasoning about qualitative conditional preferences,” J. of Heuristics, Vol. 12(4-5), pp. 263-285.

Dorigo, M. (1992). "Optimization, Learning and Natural Algorithms," Ph.D. thesis, Politecnico di Milano, Italie.

Dorigo, M. Gambardella, L. M. (1997). "Ant Colony System: A Cooperative Learning Approach to the Traveling Salesman Problem," IEEE Transactions on Evolutionary Computation, Vol. 1(1), pp. 53–66.

Dukes, M.D. Zotarelli, L. Liu, G.D. and Simonne, E.H. (2012). "Chapter 3. Principles and Practices of Irrigation Management for Vegetables," University of Florida.

Eiben, A. E. and Smith, J. E. (2003). "Introduction to evolutionary computing," 1st Edition, Springer, Natural Computing Series.

FAO. (2010). "The State of Food Insecurity in the World 2010," [Online] Available: www.fao.org.

Gajjar, C. and Joshi, J. (2011). "Socio-economic and Environmental Impact of Minor Irrigation Command Area – Case Study (A Literature Review)," National Conference on Recent Trends in Engineering & Technology, B.V.M. Engineering College.

Glover, F. (1989). "Tabu Search - Part 1," ORSA J. on Computing, Vol. 1(2), pp. 190–206.

Glover, F. (1990). "Tabu Search - Part 2," ORSA J. on Computing, Vol. 2(1), pp. 4–32.

Georgiou, P. E. and Papamichail, D. M. (2008). "Optimization model of an irrigation reservoir for water allocation and crop planning under various weather conditions," Irrigation Science, Vol. 26(6), pp. 487-504.

Grove, B. (2008). "Stochastic efficiency optimisation analysis of alternative agricultural water use strategies in Vaalharts over the long- and short-run," Ph.D. thesis, Univ. of the Free State, Bloemfontein, South Africa.

Hancock, H. (2005). "Theory of maxima and minima," Scholarly Publishing Office, University of Michigan Library.

Holland, J. H. (1975). "Adaptation in natural and artificial systems," University of Michigan Press, Ann Arbor, MI.

Jacobsen, S. E. Mujica, A. and Jensen, C. R. (2003). "The resistance of quinoa (Chenopodium quinoa Willd.) to adverse abiotic factors," Food Rev. Int. Vol. 19, pp. 99–109.

Jain, K. S and Singh, P. V. (2003). "Water Resource Systems Planning and Management," Water Resources Systems Planning and Management, Elsevier, Vol. 51, pp. 3-858.

Kantanantha, N. (2007). "Crop decision planning under yield and price uncertainties," Ph.D. thesis, Georgia Institute of Technology.

Kirkpatrick, S. Gelatt, C. D. and Vecchi, M. P. (1983). "Optimization by Simulated Annealing," Science, Vol. 220(4598), pp. 671–680.

Krauth, W. (1998). "Introduction to Monte Carlo Algorithms," Advances in Computer Simulation, Lecture Notes in Physics, Springer Verlag.

Krishnand, K. N. Ghose, D. (2009a). "Glowworm Swarm Optimisation for Simultaneous Capture of Multiple Local Optima of Multimodal Functions," Swarm Intelligence, Vol. 3, pp. 87-124.

Krishnand, K. N. Ghose, D. (2009b). "Glowworm Swarm Optimisation: A New Method for Optimizing Multimodal Functions," International J. of Computational Intelligence Studies, Vol. 1(1), pp. 93-119.

Lovewell, M. (2012). "Understanding Economics: A Contemporary Perspective," McGraw-Hill Ryerson Higher Education.

Maisela, R. J. (2010). "Realizing agricultural potential in land reform: the case of Vaalharts Irrigation Scheme in the Northern Cape Province," M. Phil mini-thesis, University of the Western Cape, Cape Town, South Africa.

Maslov B. S. "Drainage of Farmlands," Agricultural Land Improvement: Amelioration and Reclamation. Vol 2. 2009.

Mohamad, N. H. and Said, F. (2011). "A mathematical programming approach to crop mix problem," African J. of Agricultural Research, Vol. 6(1), pp. 191-197.

Mustafa, A. A. Singh, M. Sahoo, R. N. Ahmed, N. Khanna, M. Sarangi, A. and Mishra, A. K. (2011). "Land Suitability Analysis for Different Crops: A Multi Criteria Decision Making Approach using Remote Sensing and GIS," Researcher, Vol. 3(12).

Nyonyi, Y. (2010). "Modeling the hostel space allocation," Postgraduate diploma short essay, African Institute for Mathematical Sciences (AIMS), South Africa.

Oelofse, S. and Strydom, W. (2010). "A CSIR perspective on water in South Africa – 2010," CSIR Natural Resources and the Environment.

Pant, M. Radha, T. Rani, D. Abraham, A. and Srivastava, D. K. (2008). "Estimation using differential evolution for optimal crop plan," Lecture Notes in Computer Science, Vol. 5271, pp. 289-297.

Pant, M. Thangaraj, R. Rani, D. Abraham, A. and Srivastava, D. K. (2009). "Estimation of Optimal Crop Plan Using Nature Inspired Metaheuristics," World J. of Modelling and Simulation.

Pisinger, D. (1995). "Algorithms for Knapsack Problems," Ph.D Thesis, Dept. of Computer Science, University of Copenhagen.

Price, K. Storn, R. and Lampinen, A. (2005). "Differential Evolution a Practical Approach to Global Optimization," Springer Natural Computing Series.

Raju, K. S. and Kumar, N. D. (2004). "Irrigation planning using genetic algorithms", Water Resources Management, Vol. 18(2), pp. 163-176.

Raju, K. S. and Kumar, D. N. (2010). "Multicriterion Analysis in Engineering and Management," PHI Learning Pvt. Ltd. New Delhi, India, ISBN 978-81-203-3976-7, pp. 288.

Reddy, J. M. and Kumar, N. D. (2007). "Optimal reservoir operation for irrigation of multiple crops using elitist-mutated particle swarm optimization", Hydrological Sciences J., Vol. 52(4), pp. 686-701.

Rocha, M. and Neves, J. (1999). "Preventing Premature Convergence to Local Optima in Genetic Algorithms via Random Offspring Generation," Multiple Approaches to Intelligent Systems, Lecture Notes in Computer Science, Vol. 1611, pp. 127-136

Sarker, R. and Ray, T. (2009). "An improved evolutionary algorithm for solving multi-objective crop planning models," Computers and Electronics in Agriculture, Vol. 68(2), pp. 191–199.

Sarmady, S. (2012) "An Investigation on Tabu Search Parameters," University Sains Malaysia, [Online] Available: www.sarmady.com.

Schmitz, G. H. Schütze, N. and Wöhling, T. (2007). "Irrigation control: towards a new solution of an old problem," Vol. 5 of IHP/HWRP-Berichte, International Hydrological Programme (IHP) of UNESCO and The Hydrology and Water Resources Programme (HWRP) of WMO, Koblenz, Germany.

Silva, J. D. L. (2003). “Metaheuristic and multiobjective approaches for space allocation,” Ph.D. thesis. University of Nottingham.

Smook, A. J. and Project Team (2008). “Water Development and Supply Plans,” Department of Water Affairs and Forestry, Irrigation Planning and Design, [Online] Available: www.dwaf.gov.za.

Snyman, J. (2005). “Practical Mathematical Optimization: An Introduction to Basic Optimization Theory and Classical and New Gradient-Based Algorithms,” Springer.

Storn, R. Price, K. (1997). “Differential Evolution – A Simple and Efficient Heuristic for Global Optimization over Continuous Spaces,” J. of Global Optimization, Vol. 11, pp. 341–359.

Sunantara, J. D. and Ramirez, J. A. (1997). “Optimal stochastic multicrop seasonal and intraseasonal irrigation control,” J. of Water Resources Planning and Mangement, Vol. 123(1), pp. 39-48.

Syam, P, W and Al-Harkan, M. I. (2010). “Comparison of Three Meta Heurisics to Optimize Hybrid Flow Shop Scheduling Problem with Parallel Machines,” World Academy of Science, Engineering and Technology, Vol. 62.

Tan, C. M. (2008). “Simulated Annealing,” In-Tech Publisher, ISBN-13: 978-953-7619-07-7.

Trevisan, L. (2011). “Combinatorial Optimization: Exact and Approximate Algorithms,” Stanford University.

Tsai, C.Y. Wu, M.C. (2010). “Applying a Two-Stage Simulated Annealing Algorithm for Shelf Space Allocation Problems,” Proceedings of the World Congress on Engineering 2010, London, U.K., Vol. 3.

Wardlaw, R. and Bhaktikul, K. (2004). “Application of genetic algorithms for irrigation water scheduling,” Irrigation and Drainage, Vol. 53, pp. 397–414.

Weise, T. (2009). Global Optimization with Application. Online self-published electronic book.

Whelan, J. and Msefer, K. (1996). “Economic Supply & Demand,” MIT System Dynamics in Education Project.

Yang, X. S. (2010). “Nature-Inspired Metaheuristic Algorithms,” 2nd Edition, Luniver Press.

Zhao, G. Zhou, Y. and Wang, Y. (2012). “The glowworm swarm optimization algorithm with local search operator,” J. of Info. & Computational Sci. 9, Vol. 5, pp. 1299-1308.

APPENDIX A

Using the statistics of most of the crops listed in this research, 12 test datasets have been complied. These datasets have been compiled to test the ability of the algorithms in determining solutions to larger instances of ACP problems. From these 12 test datasets, 6 relates to an existing irrigation scheme and 6 to a new irrigation scheme. Similar to the evaluations in chapters 7 and 8, comparisons are made in the ability of the algorithms to determine solutions.

To determine solutions for these datasets, the heuristic specific parameter settings were set to ensure that each algorithm executed for 100,000 objective function evaluations. Each algorithm is also run 100 times. This is to determine the overall best and average performances in determining solutions.

The initial parameter settings for all metaheuristic algorithms were set to be the same as the settings found in section 8.5. The only exception is that m for LADA was set to 4. All simulations were run using the same computer system that had been used to determine solutions in sections 7.4 and 8.5.

A.1. Existing Irrigation Scheme

The hectare allocations for the datasets below consider that at existing irrigation schemes the crop planners may only be interested in determining solutions for the primary crops grown. Therefore for these datasets, the total area of land allocated to each of the summer and winter crop groups have not been set to be the same. However, if all the crops grown at an existing irrigation scheme are considered then the land allocations for the crop groups would be the same.

For the datasets at existing irrigation schemes, the crop types, the crop names, the hectares per crop (ha's crop^{-1}), the tons of yield per hectare (t ha^{-1}), the Crop Water Requirement (CWR), the average rainfall (AR), the lower and upper bounds and the producer prices (ZAR t^{-1}) for each crop has been given. Two datasets consist of a collection of 12 crops; two consist of 15 crops and two consist of 20 crops.

A.1.1. Test Dataset 1

This dataset consist of 12 crops. The total area of land allocated for the Perennial crops is 7,600 ha, the total area of land allocated for the summer crops is 18,600 ha and the total area of land allocated for the winter crops is 17,600 ha.

Table A.1: Test dataset 1

Crop Types	Crop	ha's crop^{-1}	t ha^{-1}	CWR	AR	Lower Bound	Upper Bound	ZAR t^{-1}
Perennial	Pecan Nuts	100	5.0	1,600	444.7	50	150	3,500.0
	Lucerne	7,500	16.0	1,445	444.7	7,100	7,900	1,185.52
Summer	Cotton	2,000	3.5	700	386.4	1,000	3,000	4,500.00
	Maize	6,500	9.0	979	279.0	5,000	8,000	1,321.25
	Groundnuts	7,000	3.0	912	339.5	4,500	9,500	5,076.00
	Tomato	3,000	50.0	1,132	350.8	1,500	4,000	4,332.00
	Pumpkin	100	20.0	794	279.0	50	200	1,577.09
Winter	Barley	2,200	6.0	530	58.3	1,500	4,000	2,083.27
	Wheat	12,000	6.0	650	58.3	10,000	13,000	2,174.64
	Onion	1,400	30.0	429	177.0	800	2,200	2,397.90
	Potato	1,700	28.0	365	152.8	1,000	2,700	2,463.00
	Cabbage	300	50.0	350	152.8	150	500	1,437.58

A.1.1.1. Average Execution Times

The average execution times of the algorithms in determining solutions for test dataset 1 are given in Table A.1.1.1. Table A.1.1.1 gives the statistics of the average execution times (AVG) in milliseconds (ms), and the 95% confidence interval (95% CI) values.

Table A.1.1.1: The average execution times, in milliseconds, and the 95% confidence interval values of each metaheuristic algorithm

Methods	AVG (ms)	95% CI
CS	1001	AVG $\pm$ 3.8
FA	2711	AVG $\pm$ 10.3
GSO	650	AVG $\pm$ 3.6
GA	767	AVG $\pm$ 8.0
BPA	225	AVG $\pm$ 5.2
IBPA	208	AVG $\pm$ 2.9
LADA	126	AVG $\pm$ 1.6
TS	199	AVG $\pm$ 0.9
SA	168	AVG $\pm$ 2.3

The AVG values determined by the algorithms are similar to the performances given in sections 7.4 and 8.5. A graphical representation of their performances is given in Figure A.1.1.1 below.

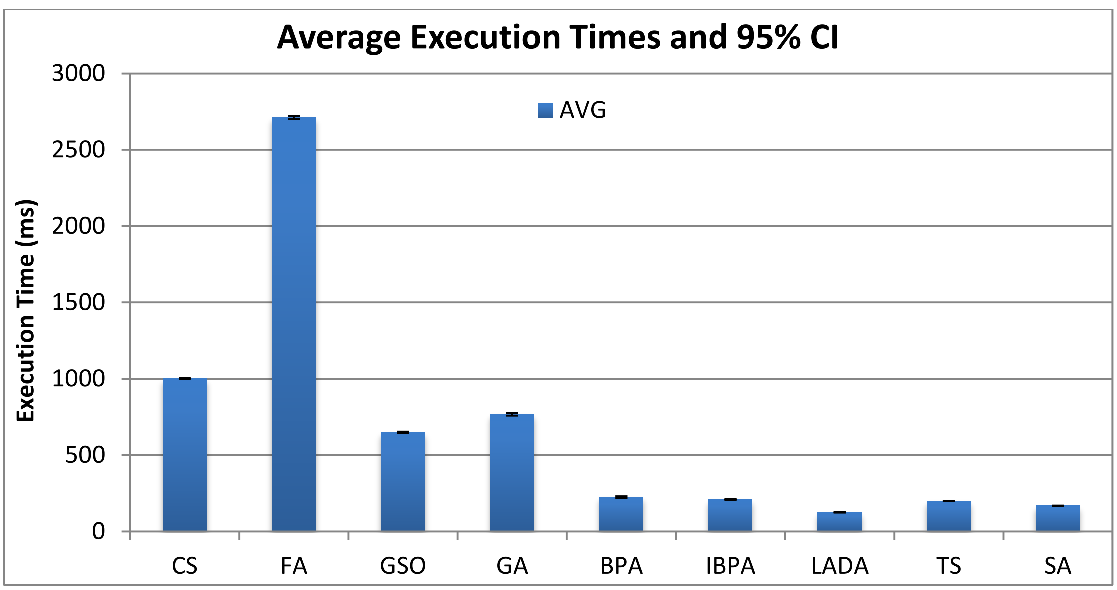

Figure A.1.1.1: The average execution times, in milliseconds (ms), and the 95% CI values of each metaheuristic algorithm

A.1.1.2. Best and Average Fitness Values

Table A.1.1.2 gives the statistical values of the overall best (BFV) and average best (ABFV) fitness values of each metaheuristic algorithm, and the BFV of the current practice (CP). The 95% CI values for the fitness value populations of each algorithm is also given.

Table A.1.1.2: Statistics for the best fitness values (BFV), average best fitness values (ABFV) and 95% confidence interval (95% CI) values

Methods	**BFV (ZAR)**	**ABFV (ZAR)**	**95% CI**
CP	932,644,726	N/A	N/A
CS	1,093,593,729	1,093,593,585	ABFV $\pm$ 20
FA	1,089,100,307	1,078,677,746	ABFV $\pm$ 1,985,478
GSO	1,088,854,007	1,028,732,217	ABFV $\pm$ 11,725,599
GA	1,074,309,904	1,047,321,084	ABFV $\pm$ 2,148,852
BPA	1,093,430,848	1,093,209,111	ABFV $\pm$ 32,235
IBPA	1,093,475,351	1,093,308,062	ABFV $\pm$ 25,378
LADA	1,093,097,520	1,092,693,187	ABFV $\pm$ 55,156
TS	1,091,022,396	1,089,811,995	ABFV $\pm$ 186,231
SA	1,048,409,914	1,032,377,083	ABFV $\pm$ 2,083,434

The BFV solution was determined by CS. The BFV performances of IBPA, BPA and LADA were marginally inferior to CS. Similarly, the best ABFV and lowest 95% CI values were also determined by CS. This was followed by IBPA, BPA and LADA. These were the four best metaheuristic algorithm performances. From all metaheuristic algorithms, CS performed the best. For the LS algorithms, IBPA performed the best. A graphical comparison of the statistical values given in Table A.1.1.2 is shown in Figure A.1.1.2 below.

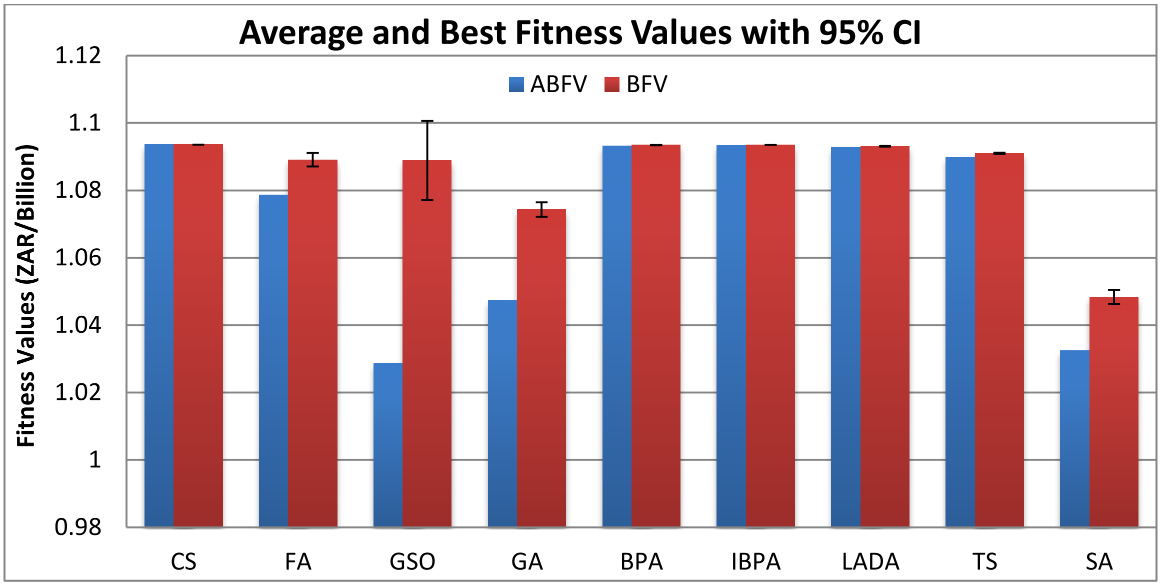

Figure A.1.1.2: A comparison of each algorithms best and average fitness values determined, along with the 95% CI estimates

A.1.1.3. Irrigated Water Requirements

Table A.1.1.3 gives the IWR's of the best solution determined by each metaheuristic algorithm, and that of the current practice (CP).

Table A.1.1.3: Statistics of the irrigated water requirements (IWR)

Methods	**IWR (m^3)**
CP	281,084,200
CS	280,520,054
FA	280,514,885
GSO	280,631,455
GA	280,281,164
BPA	280,518,356
IBPA	280,592,825
LADA	280,798,745
TS	281,731,166
SA	280,523,375

A graphical representation of the statistics given in Table A.1.1.3 is shown in Figure A.1.1.3.

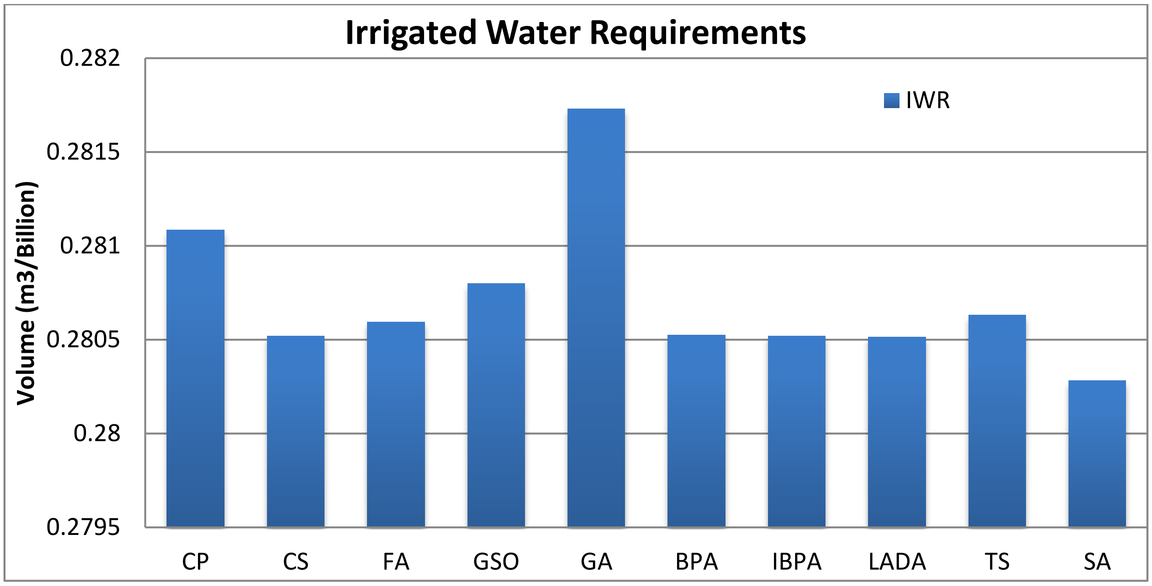

Figure A.1.1.3: Irrigated water requirements (IWR) of the best metaheuristic solutions

As can be seen from Figure A.1.1.3, the IWR for GA was higher than CP. The IWR of SA was the least. However, SAs' BFV performance was the worst overall. SAs' IWR value is therefore relative to its BFV solution found. The IWR values of CS, IBPA, BPA and LADA were similar.

A.1.1.4. Crop Hectare Allocations

Table A.1.1.4 gives the plot type hectare allocations of each crop type, as determined by the best solution of each metaheuristic algorithm.

Table A.1.1.4: Plot type hectare allocations of each crop type

Crops	Methods								
	CS	FA	GSO	GA	BPA	IBPA	LADA	TS	SA
P/Nuts	50	96	55	85	52	50	52	120	135
W/Grapes	7,550	7,504	7,545	7,515	7,548	7,550	7,548	7,480	7,465
Cotton	1,291	1,316	1,297	1,305	1,292	1,292	1,297	1,302	1,685
Maize	7,687	7,651	7,671	7,743	7,687	7,688	7,685	7,679	7,551
G/Nuts	5,812	5,781	5,833	5,784	5,811	5,811	5,810	5,806	5,692
Tomato	3,745	3,722	3,733	3,701	3,745	3,745	3,743	3,738	3,553
Pumpkin	65	130	66	68	65	65	65	76	120
Barley	1,635	1,641	1,668	1,714	1,636	1,636	1,640	1,709	2,210
Wheat	10,898	10,946	10,959	11,159	10,898	10,898	10,898	10,865	10,848
Onion	2,071	2,057	2,077	2,055	2,071	2,070	2,069	2,055	1,909
Potato	2,507	2,500	2,488	2,381	2,505	2,506	2,504	2,482	2,242
Cabbage	490	456	408	292	490	490	489	488	390

A graphical representation of the statistics given in Table A.1.1.4 is given in Figure A.1.1.4 below. The hectare allocations of each metaheuristic algorithm were similar.

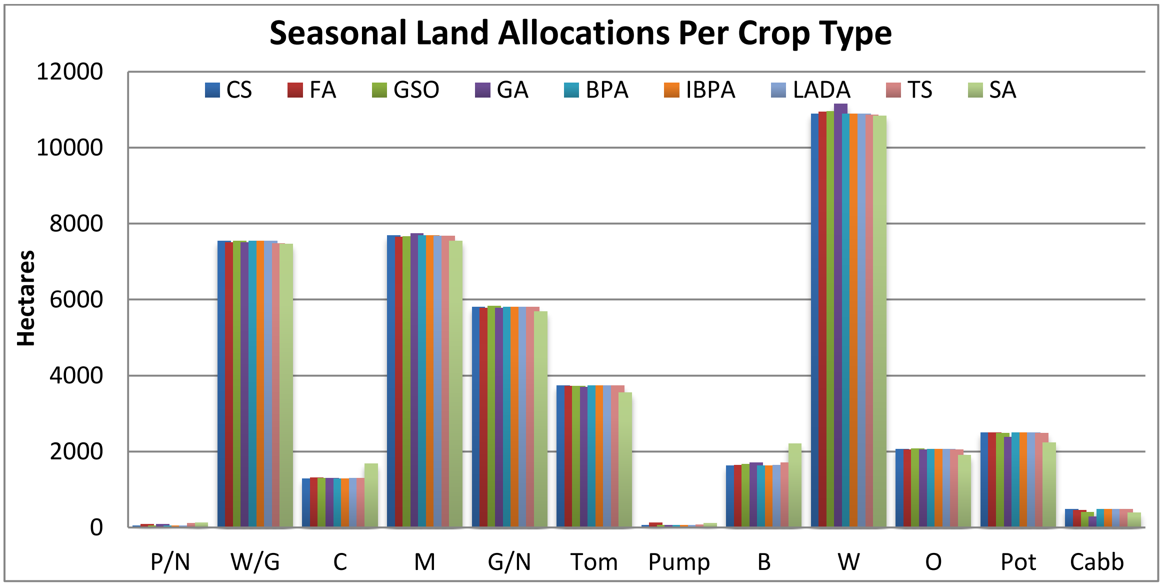

Figure A.1.1.4: A comparison of the hectare allocations, per crop, for the best solution found by each metaheuristic algorithm

A.1.2. Test Dataset 2

Similar to test dataset 1, this dataset also consists of 12 crops. This dataset is similar to test dataset 1 except for the two additional perennial crops and less summer and winter crops. The total area of land allocated for the perennial crops is 8,300 ha, the total area of land allocated for the summer crops is 18,500 ha, and the total area of land allocated for the winter crops is 17,300 ha.

Table A.2: Test dataset 2

Crop Types	Crop	ha's crop^{-1}	t ha^{-1}	CWR	AR	Lower Bound	Upper Bound	ZAR t^{-1}
Perennial	Pecan Nuts	100	5.0	1,600	444.7	50	150	3,500.0
	Wine Grapes	300	9.5	850	350.8	150	450	2,010.00
	Olives	400	6.0	1,200	444.7	250	600	2,500.00
	Lucerne	7,500	16.0	1,445	444.7	7,100	7,900	1,185.52
Summer	Cotton	2,000	3.5	700	386.4	1,000	3,000	4,500.00
	Maize	6,500	9.0	979	279.0	5,000	8,000	1,321.25
	Groundnuts	7,000	3.0	912	339.5	4,500	9,500	5,076.00
	Tomato	3,000	50.0	1,132	350.8	1,500	4,000	4,332.00
Winter	Barley	2,200	6.0	530	58.3	1,500	4,000	2,083.27
	Wheat	12,000	6.0	650	58.3	10,000	13,000	2,174.64
	Onion	1,400	30.0	429	177.0	800	2,200	2,397.90
	Potato	1,700	28.0	365	152.8	1,000	2,700	2,463.00

A.1.2.1. Average Execution Times

The average execution times of the algorithms for test dataset 2 are given in Table A.1.2.1. Table A.1.2.1 gives the statistics of the average execution times (AVG) in milliseconds (ms), and the 95% Confidence Interval (95% CI) values.

Table A.1.2.1: The average execution times, in milliseconds, and the 95% confidence interval values of each metaheuristic algorithm

Methods	AVG (ms)	95% CI
CS	992	AVG $\pm$ 3.8
FA	2,494	AVG $\pm$ 13.1
GSO	571	AVG $\pm$ 3.1
GA	743	AVG $\pm$ 7.3
BPA	212	AVG $\pm$ 1.6
IBPA	201	AVG $\pm$ 5.9
LADA	120	AVG $\pm$ 1.0
TS	193	AVG $\pm$ 1.0
SA	161	AVG $\pm$ 0.7

Table A.1.2.1 shows again that LADA executed the fastest, while FA was the slowest. A graphical representation of their performances is given in Figure A.1.2.1 below.

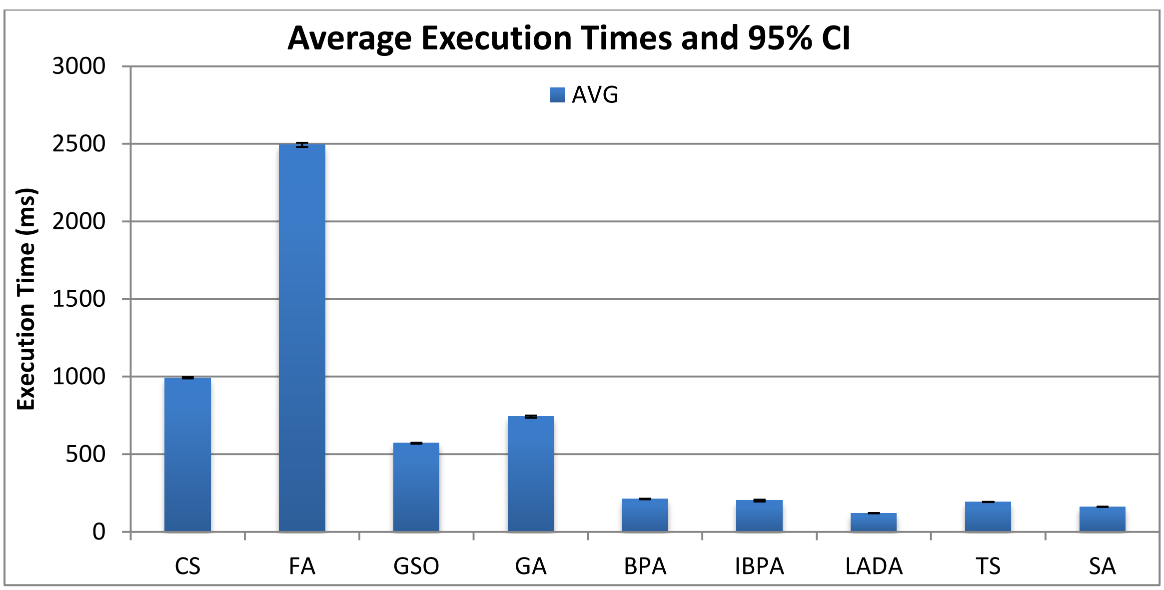

Figure A.1.2.1: The average execution times, in milliseconds (ms), and the 95% CI values of each metaheuristic algorithm

A.1.2.2. Best and Average Fitness Values

Table A.1.2.2 gives the statistical values of the BFV and ABFV values of each metaheuristic algorithm. It also gives the BFV of CP and the 95% CI fitness values.

Table A.1.2.2: Statistics for the best fitness values (BFV), average best fitness values (ABFV) and 95% confidence interval (95% CI) values

Methods	BFV (ZAR)	ABFV (ZAR)	95% CI
CP	923,685,834	N/A	N/A
CS	1,078,861,903	1,078,861,794	ABFV ± 18
FA	1,076,311,407	1,067,900,015	ABFV ± 2,080,747
GSO	1,075,346,901	1,021,880,654	ABFV ± 10,123,234
GA	1,052,599,353	1,035,986,311	ABFV ± 1,828,358
BPA	1,078,719,028	1,078,570,131	ABFV ± 31,326
IBPA	1,078,772,261	1,078,598,868	ABFV ± 26,436
LADA	1,077,928,251	1,077,124,575	ABFV ± 108,083
TS	1,077,687,887	1,076,653,892	ABFV ± 153,602
SA	1,037,500,145	1,023,397,224	ABFV ± 2,257,393

Similar to test dataset 1, CS, IBPA, BPA and LADA were the four best metaheuristic algorithms. A graphical comparison of the statistical values given in Table A.1.2.2 is shown in Figure A.1.2.2 below.

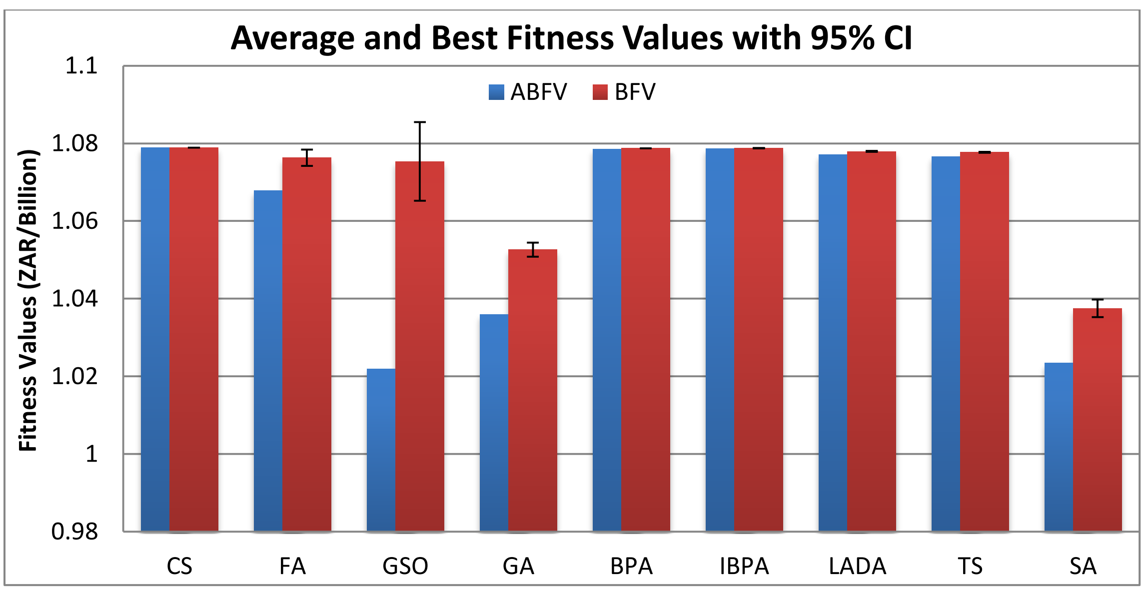

Figure A.1.2.2: A comparison of each algorithms best and average fitness values determined, along with the 95% CI estimates

A.1.2.3. Irrigated Water Requirements

Table A.1.2.3 gives the IWR's of the best solution determined by each metaheuristic algorithm, and that of CP.

Table A.1.2.3: Statistics of the irrigated water requirements (IWR)

Methods	IWR (m^3)
CP	284,496,400
BPA	284,059,343
IBPA	284,095,711
LADA	285,130,270
TS	285,773,881
SA	284,057,441
CS	284,055,470
FA	284,069,535
GSO	283,990,221
GA	285,656,899

A graphical representation of the statistics given in Table A.1.2.3 is shown in Figure A.1.2.3 below.

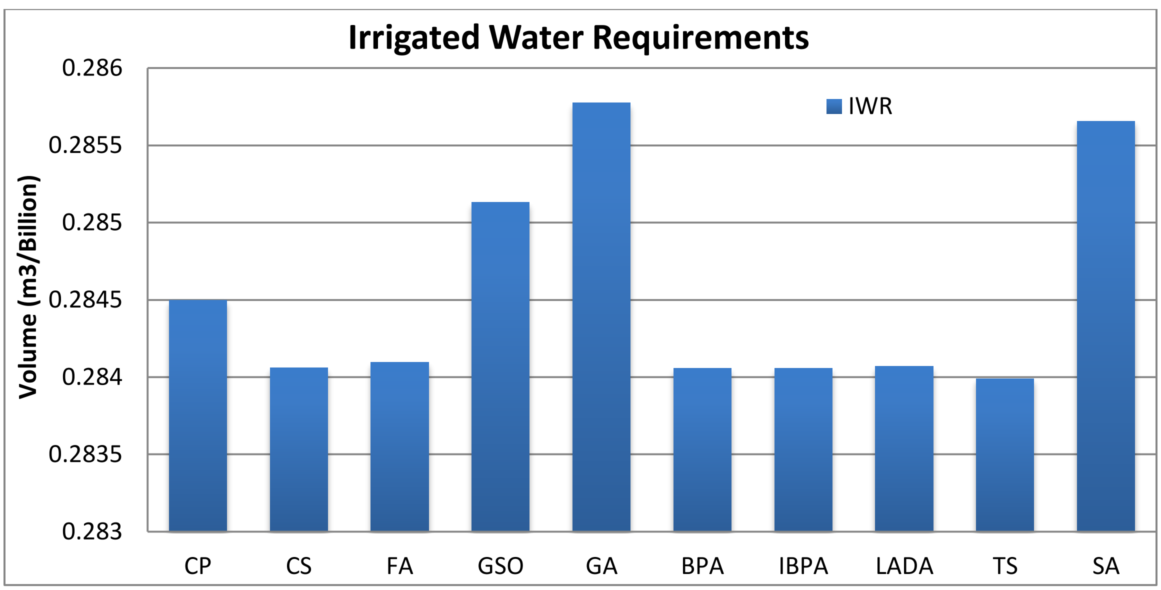

Figure A.1.2.3: Irrigated water requirements (IWR) of the best metaheuristic solutions

As can be seen from Figure A.1.2.3, the IWR for GSO, GA and SA was higher than CP. This proves that these algorithms did not determine good solutions. The IWR values of CS, FA, IBPA, BPA, LADA and TS were similar.

A.1.2.4. Crop Hectare Allocations

Table A.1.2.4 gives the plot type hectare allocations of the best solution determined by each metaheuristic algorithm.

Table A.1.2.4: Plot type hectare allocations of each crop type

Crops	Methods								
	CS	**FA**	**GSO**	**GA**	**BPA**	**IBPA**	**LADA**	**TS**	**SA**
P/Nuts	50	66	94	131	50	50	52	65	59
W/Grapes	429	345	204	177	429	430	430	425	289
Olives	286	472	347	350	287	287	291	330	498
Lucerne	7,535	7,417	7,655	7,642	7,534	7,532	7,527	7,479	7,453
Cotton	1,289	1,309	1,300	1,410	1,290	1,289	1,291	1,293	1,385
Maize	7,672	7,663	7,689	7,564	7,673	7,672	7,673	7,673	7,886
G/Nuts	5,801	5,795	5,792	5,869	5,800	5,801	5,801	5,802	5,664
Tomato	3,738	3,733	3,718	3,657	3,738	3,738	3,734	3,732	3,565
Barley	1,653	1,664	1,688	1,945	1,654	1,654	1,656	1,653	1,580
Wheat	11,019	11,043	11,016	11,124	11,019	11,020	11,027	11,024	11,541
Onion	2,094	2,077	2,077	1,959	2,092	2,093	2,089	2,092	1,960
Potato	2,534	2,515	2,519	2,272	2,534	2,534	2,529	2,531	2,218

A graphical representation of the statistics given in Table A.1.2.4 is given in Figure A.1.2.4 below. The hectare allocations of each metaheuristic algorithm were again similar.

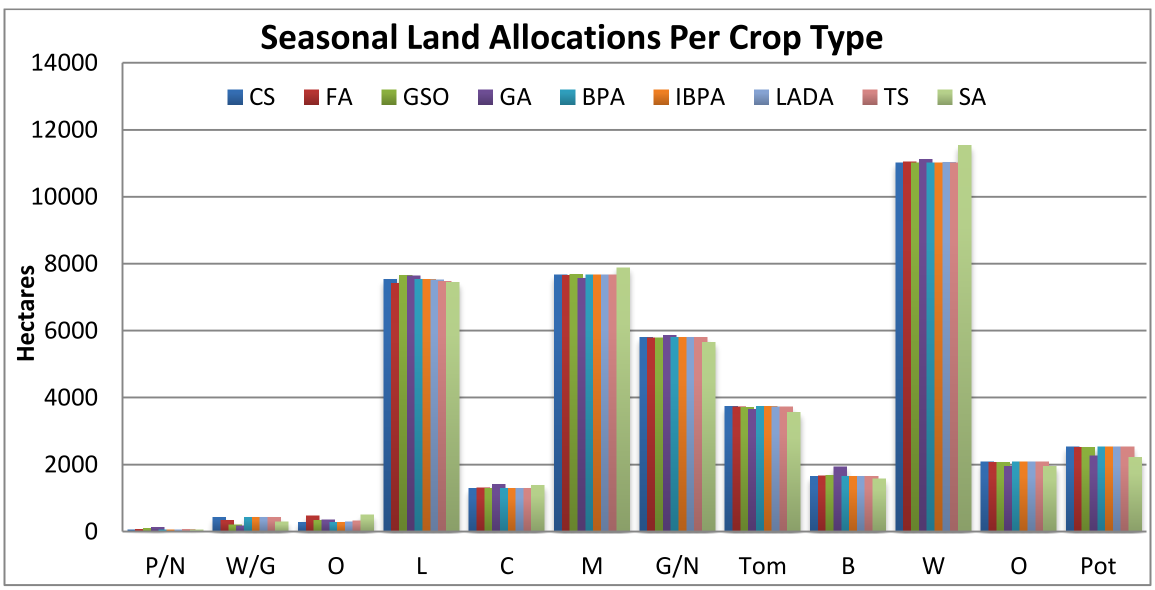

Figure A.1.2.4: A comparison of the hectare allocations, per crop, for the best solution found by each metaheuristic algorithm

A.1.3. Test Dataset 3

This dataset consists of 15 crops. It is a combination of all the crops listed in test datasets 1 and 2. The total area of land allocated for the Perennial crops is 8,300 ha, the total area of land allocated for the summer crops is 19,800 ha and the total area of land allocated for the winter crops is 17,600 ha.

Table A.3: Test dataset 3

Crop Types	Crop	ha's crop^{-1}	t ha^{-1}	CWR	AR	Lower Bound	Upper Bound	ZAR t^{-1}
Perennial	Pecan Nuts	100	5.0	1,600	444.7	50	150	3,500.0
	Wine Grapes	300	9.5	850	350.8	150	450	2,010.00
	Olives	400	6.0	1,200	444.7	250	600	2,500.00
	Lucerne	7,500	16.0	1,445	444.7	7,100	7,900	1,185.52
Summer	Cotton	2,000	3.5	700	386.4	1,000	3,000	4,500.00
	Maize	6,500	9.0	979	279.0	5,000	8,000	1,321.25
	Groundnuts	7,000	3.0	912	339.5	4,500	9,500	5,076.00
	Tomato	3,000	50.0	1,132	350.8	1,500	4,000	4,332.00
	Pumpkin	100	20.0	794	279.0	50	200	1,577.09
	Sunflower	1200	3.0	648	314.9	600	1,800	3,739.00
Winter	Barley	2,200	6.0	530	58.3	1,500	4,000	2,083.27
	Wheat	12,000	6.0	650	58.3	10,000	13,000	2,174.64
	Onion	1,400	30.0	429	177.0	800	2,200	2,397.90
	Potato	1,700	28.0	365	152.8	1,000	2,700	2,463.00
	Cabbage	300	50.0	350	152.8	150	500	1,437.58

A.1.3.1. Average Execution Times

The average execution times of the algorithms in determining solutions for test dataset 3 are given in Table A.1.3.1.

Table A.1.3.1: The average execution times, in milliseconds, and the 95% confidence interval values of each metaheuristic algorithm

Methods	AVG (ms)	95% CI
CS	1,305	AVG ± 47.9
FA	3,362	AVG ± 72.6
GSO	805	AVG ± 16.8
GA	893	AVG ± 29.1
BPA	272	AVG ± 3.8
IBPA	257	AVG ± 3.6
LADA	163	AVG ± 2.5
TS	253	AVG ± 4.9
SA	213	AVG ± 9.6

Table A.1.3.1 shows again that LADA was the fastest and that FA was the slowest. A graphical representation of their performances is given in Figure A.1.3.1 below.

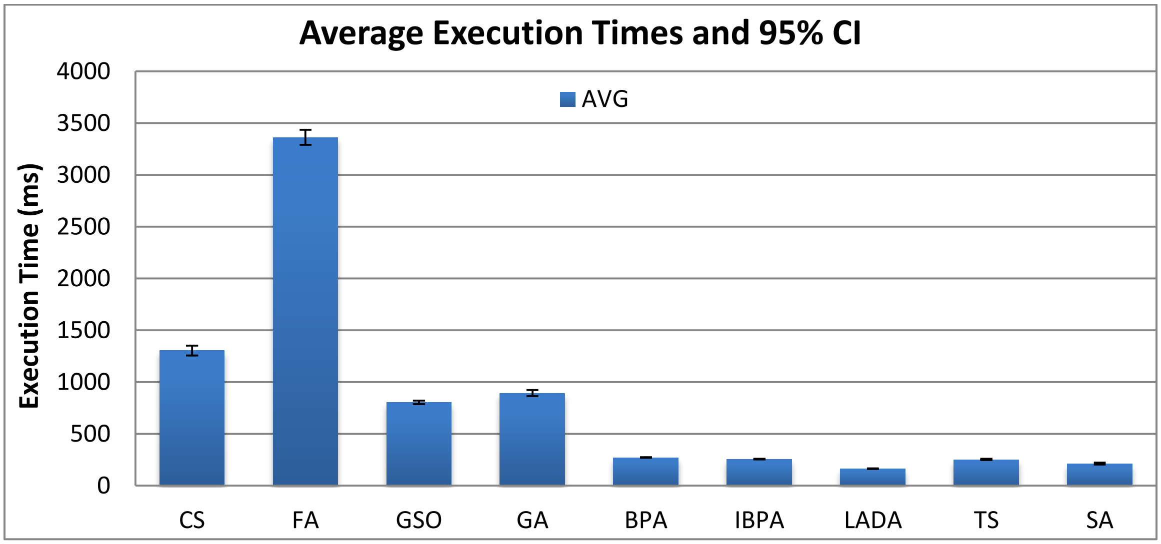

Figure A.1.3.1: The average execution times, in milliseconds (ms), and the 95% CI values of each metaheuristic algorithm

A.1.3.2. Best and Average Fitness Values

Table A.1.3.2 gives the statistical values of the fitness and 95% CI fitness values of each metaheuristic algorithm, and that of the CP.

Table A.1.3.2: Statistics for the best fitness values (BFV), average best fitness values (ABFV) and 95% confidence interval (95% CI) values

Methods	BFV (ZAR)	ABFV (ZAR)	95% CI
CP	948,690,492	N/A	N/A
CS	1,121,643,769	1,121,642,330	ABFV ± 184
FA	1,117,550,965	1,106,183,360	ABFV ± 2,201,562
GSO	1,108,573,005	1,025,916,698	ABFV ± 13,763,282
GA	1,079,589,902	1,062,741,287	ABFV ± 1,759,432
BPA	1,121,408,488	1,121,031,763	ABFV ± 57,124
IBPA	1,121,343,275	1,121,092,026	ABFV ± 42,359
LADA	1,120,322,595	1,119,250,835	ABFV ± 140,951
TS	1,118,820,894	1,116,469,884	ABFV ± 262,757
SA	1,069,593,783	1,046,064,917	ABFV ± 2,501,619

Again CS, IBPA, BPA and LADA were the four best metaheuristic algorithms. A graphical comparison of the statistical values given in Table A.1.3.2 is shown in Figure A.1.3.2 below.

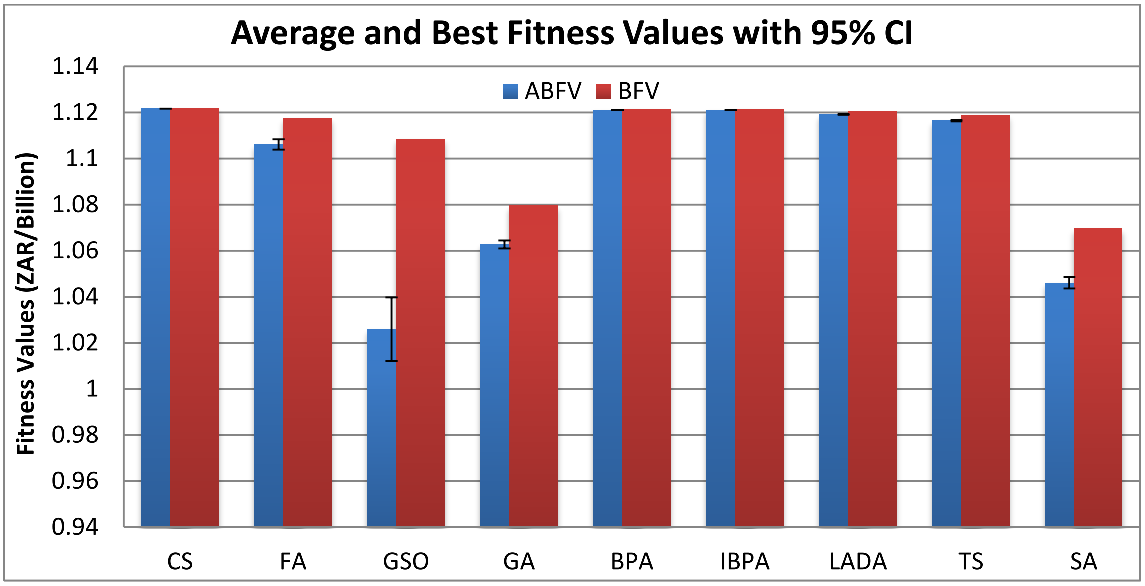

Figure A.1.3.2: A comparison of each algorithms best and average fitness values determined, along with the 95% CI estimates

A.1.3.3. Irrigated Water Requirements

Table A.1.3.3 gives the IWR's of each metaheuristic algorithm and that of CP.

Table A.1.3.3: Statistics of the irrigated water requirements (IWR)

Methods	IWR (m^3)
CP	289,600,200
BPA	289,955,298
IBPA	289,617,992
LADA	292,036,299
TS	290,579,543
SA	290,008,149
CS	289,952,678
FA	289,991,868
GSO	290,045,017
GA	290,465,056

A graphical representation of the statistics given in Table A.1.3.3 is shown in Figure A.1.3.3 below.

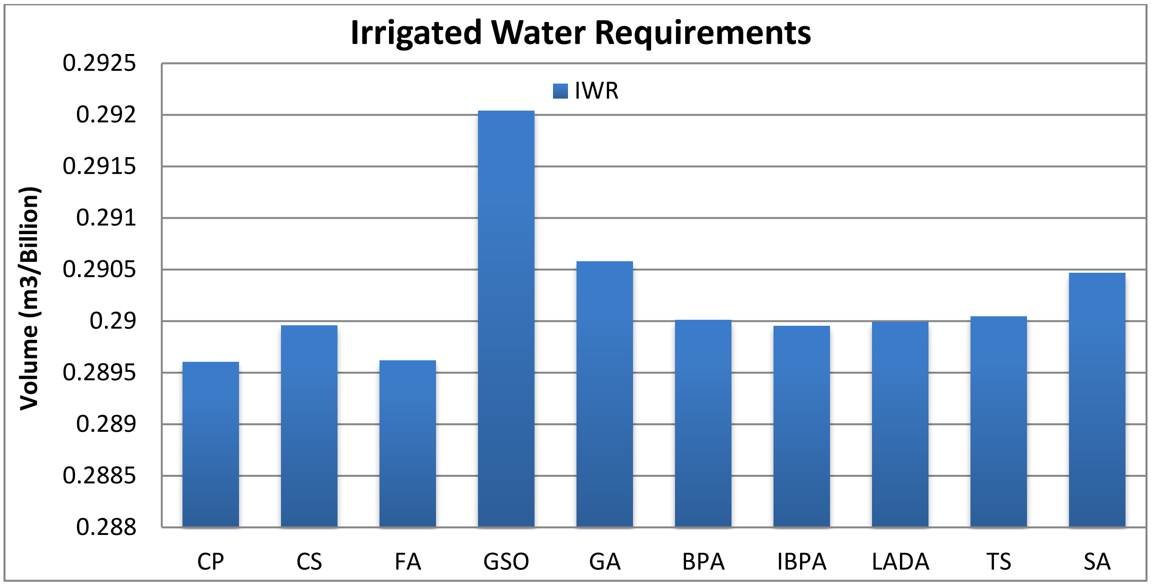

Figure A.1.3.3: Irrigated water requirements (IWR) of the best metaheuristic solutions

Although the algorithms BFV's were higher than that of the CP, Figure A.1.3.3 shows that the IWR values of each algorithm were also higher than that of the CP. Of these values, GSO, GA and SA were again the highest. The IWR values for CS, BPA, IBPA, LADA and TS were again similar.

A.1.3.4. Crop Hectare Allocations

Table A.1.3.4 gives the plot type hectare allocations of the best solution determined by each metaheuristic algorithm.

Table A.1.3.4: Plot type hectare allocations of each crop type

Crops	Methods								
	CS	FA	GSO	GA	BPA	IBPA	LADA	TS	SA
P/Nuts	50	85	56	77	52	54	55	111	78
W/Grapes	429	399	156	236	418	430	428	360	343
Olives	286	497	310	492	289	287	288	458	437
Lucerne	7,535	7,319	7,779	7,494	7,541	7,529	7,528	7,371	7,443
Cotton	1,320	1,335	1,349	1,583	1,319	1,320	1,324	1,320	1,394
Maize	7,856	7,848	7,835	7,601	7,855	7,853	7,857	7,860	7,596
G/Nuts	5,939	5,943	5,931	5,917	5,939	5,941	5,938	5,941	6,404
Tomato	3,827	3,813	3,814	3,681	3,826	3,826	3,822	3,818	3,583
Pumpkin	66	69	80	79	67	66	67	66	66
S/Flower	792	791	791	939	793	794	792	794	757
Barley	1,635	1,723	1,833	1,899	1,636	1,636	1,645	1,647	2,102
Wheat	10,898	10,852	10,984	11,202	10,898	10,899	10,903	10,913	10,934
Onion	2,071	2,058	2,085	1,859	2,070	2,069	2,070	2,065	2,001

Potato	2,506	2,491	2,525	2,372	2,506	2,506	2,492	2,496	2,296
Cabbage	490	476	174	267	490	490	489	478	268

A graphical representation of the statistical values given in Table A.1.3.4 is shown in Figure A.1.3.4 below.

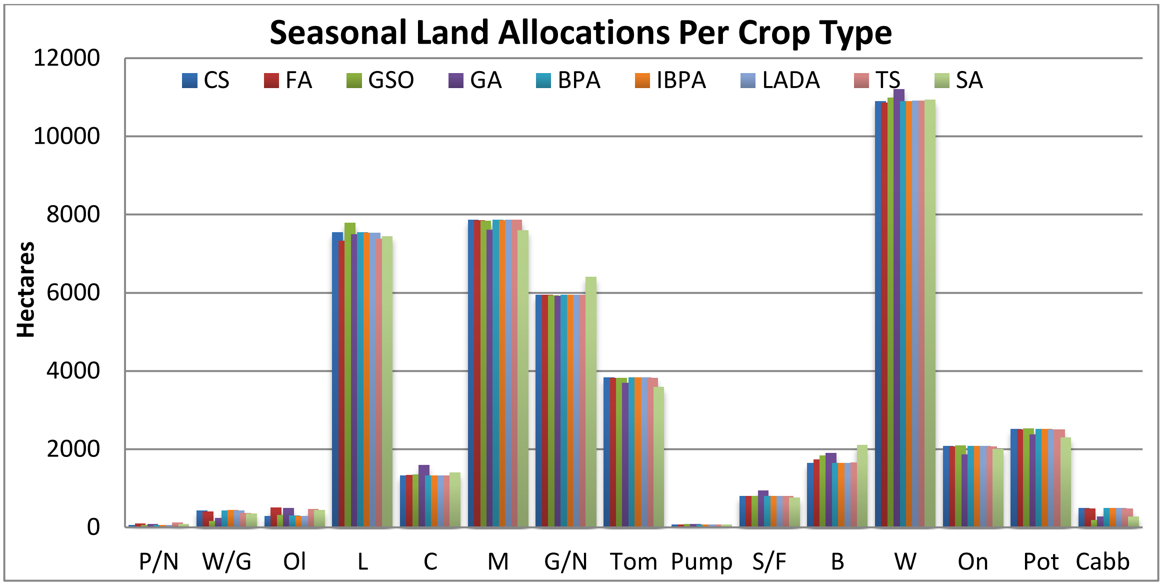

Figure A.1.3.4: A comparison of the hectare allocations, per crop, for the best solution found by each metaheuristic algorithm

A.1.4. Test Dataset 4

This dataset also consists of 15 crops. The list is the same as test dataset 3. However, the hectare allocations of each crop were set differently. The total area of land allocated for the Perennial crops is 7,300 ha, the total area of land allocated for the summer crops is 17,400 ha and the total area of land allocated for the winter crops is 18,600 ha.

Table A.4: Test dataset 4

Crop Types	Crop	ha's crop^{-1}	t ha^{-1}	CWR	AR	Lower Bound	Upper Bound	ZAR t^{-1}
Perennial	Pecan Nuts	1,000	5.0	1,600	444.7	500	1,500	3,500.0
	Wine Grapes	2,300	9.5	850	350.8	1,500	3,500	2,010.00
	Olives	2,500	6.0	1,200	444.7	1,800	3,800	2,500.00
	Lucerne	1,500	16.0	1,445	444.7	500	3,000	1,185.52
Summer	Cotton	500	3.5	700	386.4	250	800	4,500.00
	Maize	9,500	9.0	979	279.0	7,000	12,000	1,321.25
	Groundnuts	1,500	3.0	912	339.5	1,000	3,000	5,076.00
	Tomato	500	50.0	1,132	350.8	250	800	4,332.00
	Pumpkin	1,200	20.0	794	279.0	450	2,000	1,577.09
	Sunflower	4,200	3.0	648	314.9	3,200	5,800	3,739.00
Winter	Barley	7,200	6.0	530	58.3	5,800	9,500	2,083.27
	Wheat	2,000	6.0	650	58.3	1,200	3,000	2,174.64
	Onion	3,400	30.0	429	177.0	2,300	4,500	2,397.90
	Potato	2,700	28.0	365	152.8	1,900	3,800	2,463.00
	Cabbage	3,300	50.0	350	152.8	2,500	5,000	1,437.58

A.1.4.1. Average Execution Times

The average execution times and the 95% CI values are given in Table A.1.4.1 below.

Table A.1.4.1: The average execution times, in milliseconds, and the 95% confidence interval values of each metaheuristic algorithm

Methods	AVG (ms)	95% CI
CS	1,306	AVG ± 18.7
FA	3,415	AVG ± 44.1
GSO	823	AVG ± 16.9
GA	913	AVG ± 19.6
BPA	280	AVG ± 5.4
IBPA	262	AVG ± 4.2
LADA	166	AVG ± 3.2
TS	259	AVG ± 7.6
SA	215	AVG ± 4.8

A graphical representation of the values given in Table A.1.4.1 is shown in Figure A.1.4.1 below.

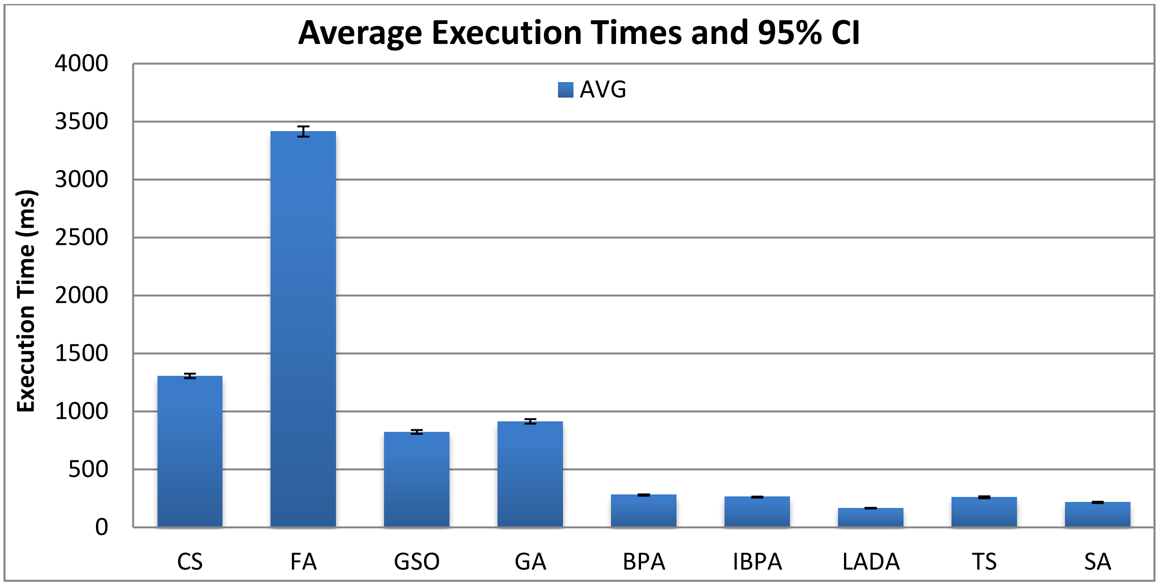

Figure A.1.4.1: The average execution times, in milliseconds (ms), and the 95% CI values of each metaheuristic algorithm

A.1.4.2. Best and Average Fitness Values

Table A.1.4.2 gives the statistical values of the BFV and ABFV values of each metaheuristic algorithm. It also gives the BFV of CP and the 95% CI fitness values.

Table A.1.4.2: Statistics for the best fitness values (BFV), average best fitness values (ABFV) and 95% confidence interval (95% CI) values

Methods	**BFV (ZAR)**	**ABFV (ZAR)**	**95% CI**
CP	812,599,360	N/A	N/A
CS	940,816,290	940,812,035	ABFV ± 701
FA	936,533,041	923,880,167	ABFV ± 2,023,049
GSO	918,842,773	874,453,781	ABFV ± 6,128,663
GA	910,728,635	900,693,053	ABFV ± 1,192,756
BPA	940,360,275	939,978,717	ABFV ± 51,048
IBPA	940,430,497	940,170,928	ABFV ± 39,239
LADA	939,915,505	939,418,676	ABFV ± 61,359
TS	937,803,353	935,478,367	ABFV ± 183,131
SA	904,795,579	894,524,615	ABFV ± 1,461,699

Again CS, IBPA, BPA and LADA were the best metaheuristic algorithms. A graphical comparison of the statistical values given in Table A.1.4.2 is shown in Figure A.1.4.2 below.

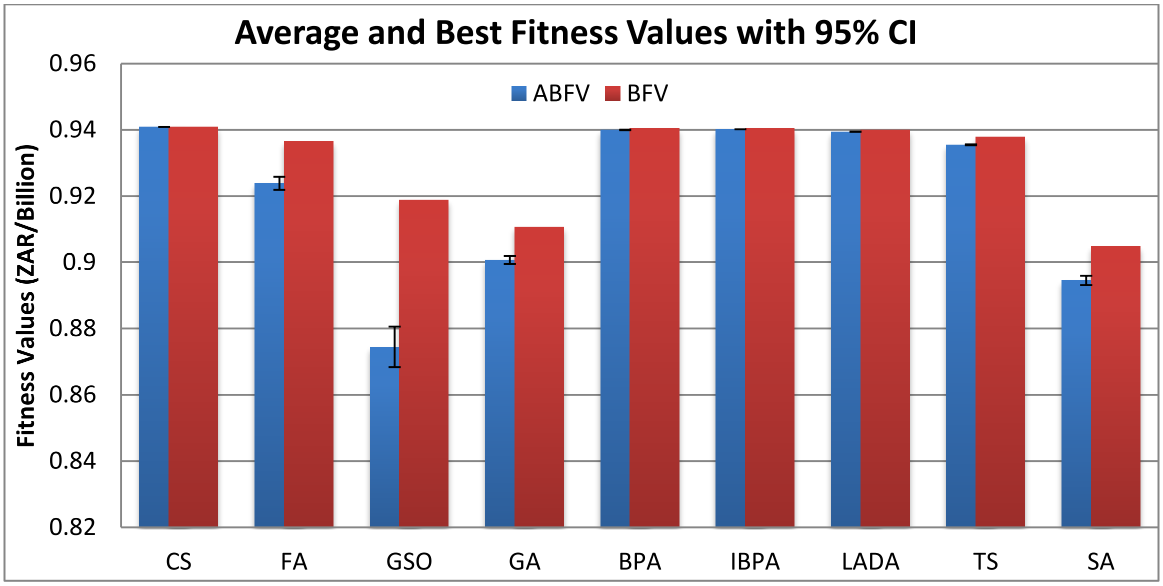

Figure A.1.4.2: A comparison of each algorithms best and average fitness values determined, along with the 95% CI estimates

A.1.4.3. Irrigated Water Requirements

Table A.1.4.3 gives the IWR's of each metaheuristic algorithm, and that of the CP.

Table A.1.4.3: Statistics of the irrigated water requirements (IWR)

Methods	IWR (m^3)
CP	224,254,700
BPA	216,653,524
IBPA	212,214,084
LADA	220,698,821
TS	213,763,196
SA	216,789,120
CS	216,656,080
FA	216,730,843
GSO	215,883,272
GA	216,834,920

A graphical representation of the statistics given in Table A.1.4.3 is shown in Figure A.1.4.3 below.

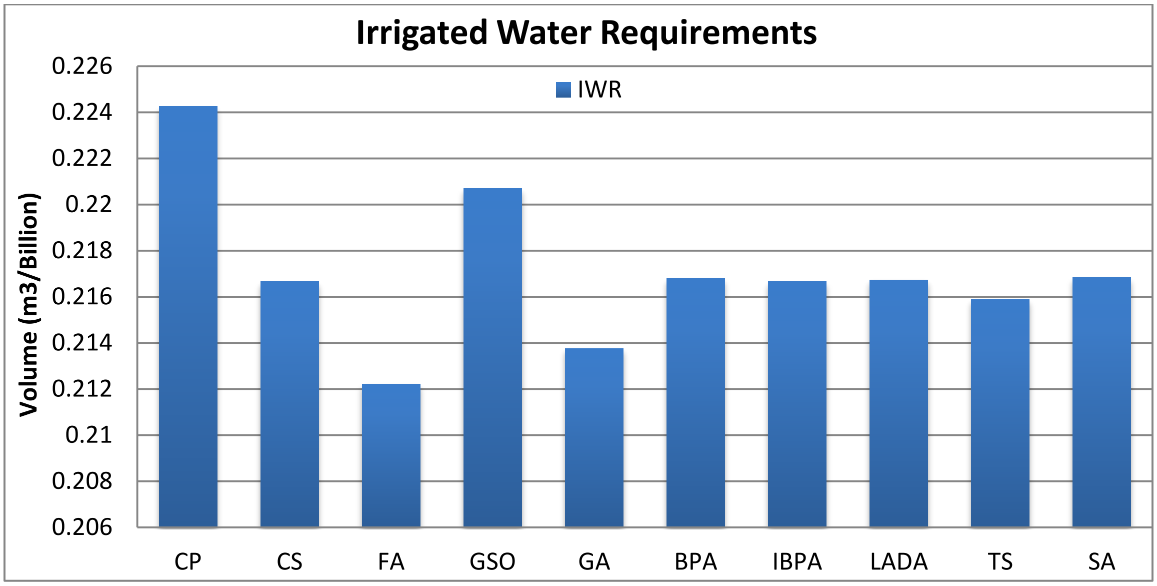

Figure A.1.4.3: Irrigated water requirements (IWR) of the best metaheuristic solutions

Figure A.1.4.3 shows that all metaheuristic algorithms determined improved IWR values.

A.1.4.4. Crop Hectare Allocations

Table A.1.4.4 gives the plot type hectare allocations of each crop type as determined by the best solution of each metaheuristic algorithm.

Table A.1.4.4: Plot type hectare allocations of each crop type

Crops	**Methods**								
	CS	**FA**	**GSO**	**GA**	**BPA**	**IBPA**	**LADA**	**TS**	**SA**
P/Nuts	503	838	1169	800	532	511	521	611	698
W/Grapes	2,642	3,258	1,611	2,583	2,621	2,642	2,625	2,796	1,716
Olives	1,953	2,666	2,357	2,298	1,961	1,955	1,965	2,084	2,809
Lucerne	2,202	539	2,163	1,618	2,185	2,193	2,188	1,810	2,076
Cotton	325	425	469	819	326	335	330	346	476
Maize	9,110	9,068	9,324	8,458	9,113	9,105	9,098	9,166	8,077
G/Nuts	1,301	1,331	1,396	1,338	1,302	1,303	1,309	1,306	1,804
Tomato	742	736	745	662	742	741	740	739	625
Pumpkin	1,757	1,698	461	1,165	1,745	1,754	1,749	1,677	1,404
S/Flower	4,165	4,141	5,005	4,957	4,172	4,163	4,175	4,166	5,014
Barley	5,807	5,823	5,872	5,911	5,812	5,813	5,813	5,830	6,048
Wheat	1,179	1,198	1,204	1,374	1,180	1,180	1,186	1,190	1,311
Onion	3,930	3,909	3,772	3,918	3,925	3,928	3,928	3,929	3,901
Potato	3,318	3,307	3,350	3,209	3,319	3,312	3,318	3,273	2,981
Cabbage	4,366	4,363	4,402	4,188	4,365	4,366	4,355	4,379	4,358

A graphical representation of the statistical values given in Table A.1.4.4 is shown in Figure A.1.4.4 below.

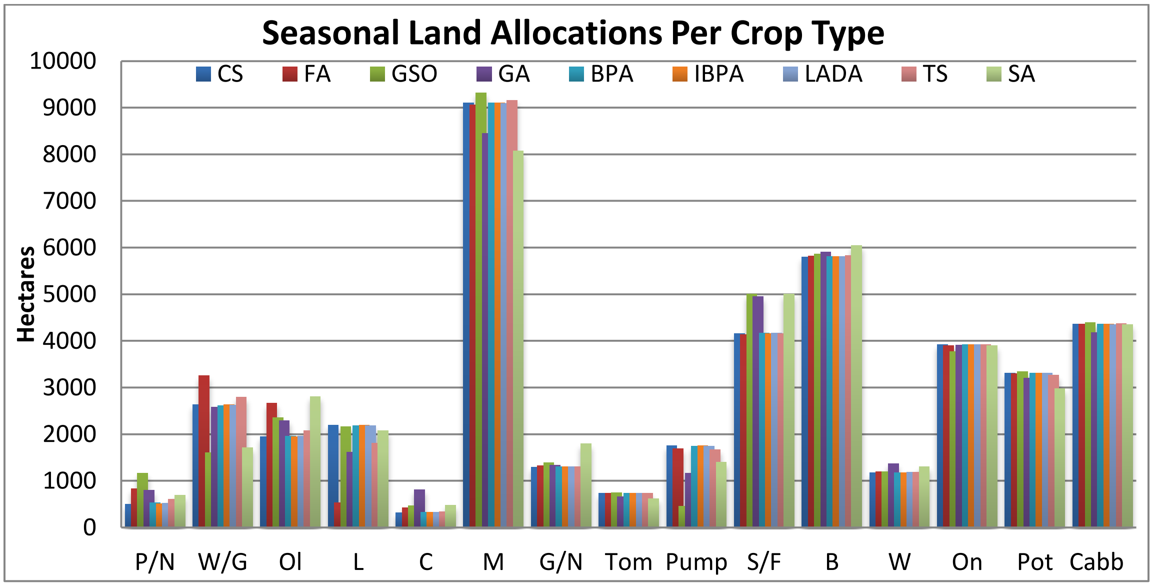

Figure A.1.4.4: A comparison of the hectare allocations, per crop, for the best solution found by each metaheuristic algorithm

A.1.5. Test Dataset 5

This dataset consists of 20 crops. It extends test dataset 3 by adding an additional 5 crops. The total area of land allocated for the Perennial crops is 8,300 ha, the total area of land allocated for the summer crops is 20,150 ha and the total area of land allocated for the winter crops is 19,300 ha.

Table A.5: Test dataset 5

Crop Types	Crop	ha's crop^{-1}	t ha^{-1}	CWR	AR	Lower Bound	Upper Bound	ZAR t^{-1}
Perennial	Pecan Nuts	100	5.0	1,600	444.7	50	150	3,500.0
	Wine Grapes	300	9.5	850	350.8	150	450	2,010.00
	Olives	400	6.0	1,200	444.7	250	600	2,500.00
	Lucerne	7,500	16.0	1,445	444.7	7,100	7,900	1,185.52
Summer	Cotton	2,000	3.5	700	386.4	1,000	3,000	4,500.00
	Maize	6,500	9.0	979	279.0	5,000	8,000	1,321.25
	Groundnuts	7,000	3.0	912	339.5	4,500	9,500	5,076.00
	Tomato	3,000	50.0	1,132	350.8	1,500	4,000	4,332.00
	Pumpkin	100	20.0	794	279.0	50	200	1,577.09
	Sunflower	1200	3.0	648	314.9	600	1,800	3,739.00
	Dry Beans	200	2.0	650	269.2	100	400	5,600.00
	Soya Beans	150	3.0	600	269.2	50	350	2,528.01
Winter	Barley	2,200	6.0	530	58.3	1,500	4,000	2,083.27
	Wheat	12,000	6.0	650	58.3	10,000	13,000	2,174.64
	Onion	1,400	30.0	429	177.0	800	2,200	2,397.90
	Potato	1,700	28.0	365	152.8	1,000	2,700	2,463.00
	Cabbage	300	50.0	350	152.8	150	500	1,437.58
	Water Melon	500	20.0	500	22.4	350	700	934.00
	Cauliflower	400	10.0	500	152.8	300	600	4,252.00
	lettuce	800	20.0	300	33.7	500	1,500	4,432.00

A.1.5.1. Average Execution Times

The average execution times of the algorithms are given in Table A.1.5.1 below.

Table A.1.5.1: The average execution times, in milliseconds, and the 95% confidence interval values of each metaheuristic algorithm

Methods	AVG (ms)	95% CI
CS	1,643	AVG ± 5.8
FA	3,968	AVG ± 16.2
GSO	817	AVG ± 17.2
GA	1,020	AVG ± 18.7
BPA	336	AVG ± 2.3
IBPA	317	AVG ± 2.8
LADA	197	AVG ± 1.4
TS	312	AVG ± 1.3
SA	260	AVG ± 3.2

A graphical representation of the algorithms performances is shown in Figure A.1.5.1 below.

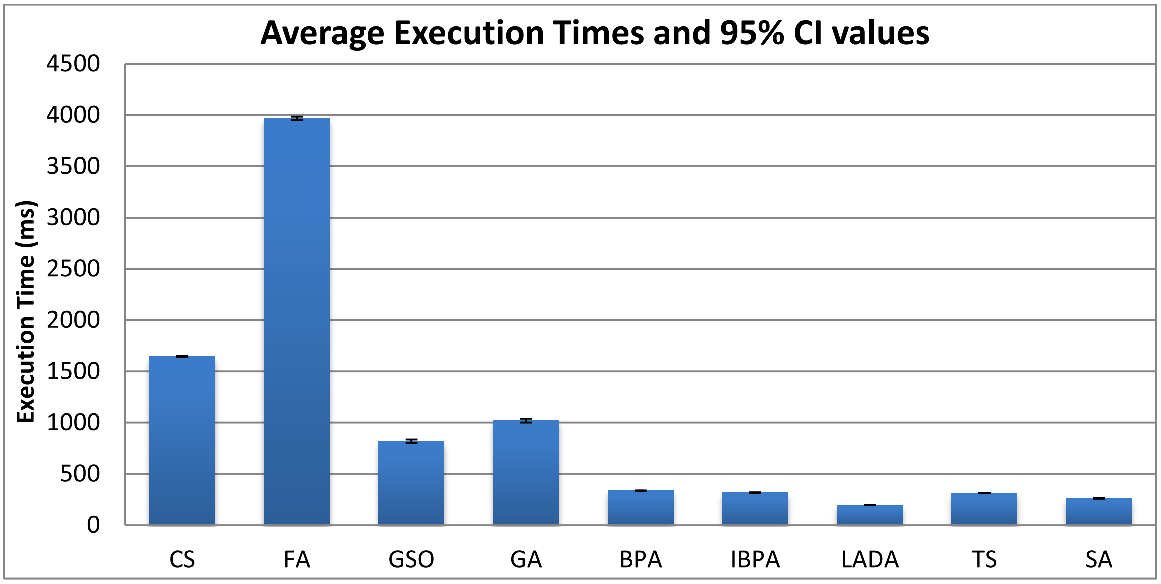

Figure A.1.5.1: The average execution times, in milliseconds (ms), and the 95% CI values of each metaheuristic algorithm

A.1.5.2. Best and Average Fitness Values

Table A.1.5.2 gives the statistical values of the BFV and ABFV values of each metaheuristic algorithm. It also gives the BFV of the CP and the 95% CI fitness values.

Table A.1.5.2: Statistics for the best fitness values (BFV), average best fitness values (ABFV) and 95% confidence interval (95% CI) values

Methods	BFV (ZAR)	ABFV (ZAR)	95% CI
CP	1,015,153,957	N/A	N/A
CS	1,217,097,419	1,217,074,709	ABFV ± 2,996
FA	1,210,133,562	1,193,402,878	ABFV ± 2,894,260
GSO	1,147,040,976	1,053,749,462	ABFV ± 14,629,463
GA	1,157,164,650	1,136,633,718	ABFV ± 2,111,637
BPA	1,216,093,121	1,215,240,526	ABFV ± 125,491
IBPA	1,216,159,829	1,215,476,100	ABFV ± 93,595
LADA	1,214,984,543	1,213,835,287	ABFV ± 137,531
TS	1,209,300,640	1,204,735,213	ABFV ± 403,786
SA	1,141,774,391	1,121,525,585	ABFV ± 2,497,189

A graphical comparison of the statistical values given in Table A.1.5.2 is shown in Figure A.1.5.2 below.

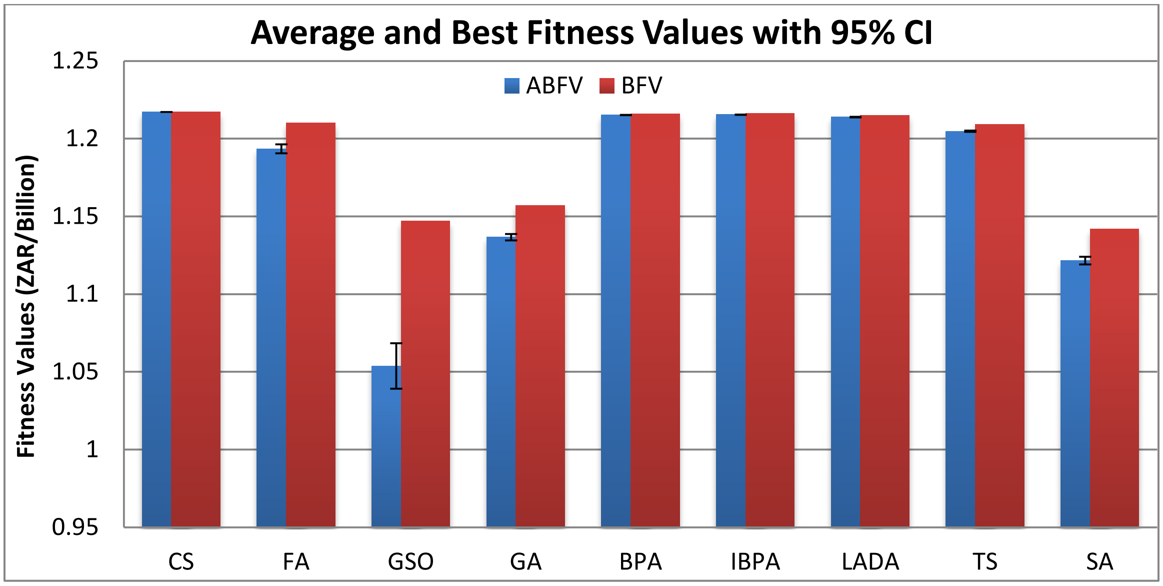

Figure A.1.5.2: A comparison of each algorithms best and average fitness values determined, along with the 95% CI estimates

A.1.5.3. Irrigated Water Requirements

Table A.1.5.3 gives the IWR's of each metaheuristic algorithm, and that of the CP.

Table A.1.5.3: Statistics of the irrigated water requirements (IWR)

Methods	**IWR (m^3)**
CP	296,765,200
BPA	296,197,908
IBPA	295,855,625
LADA	302,656,627
TS	297,196,010
SA	296,209,781
CS	296,330,663
FA	296,125,157
GSO	296,703,395
GA	298,554,213

A graphical representation of the statistics given in Table A.1.5.3 is shown in Figure A.1.5.3 below.

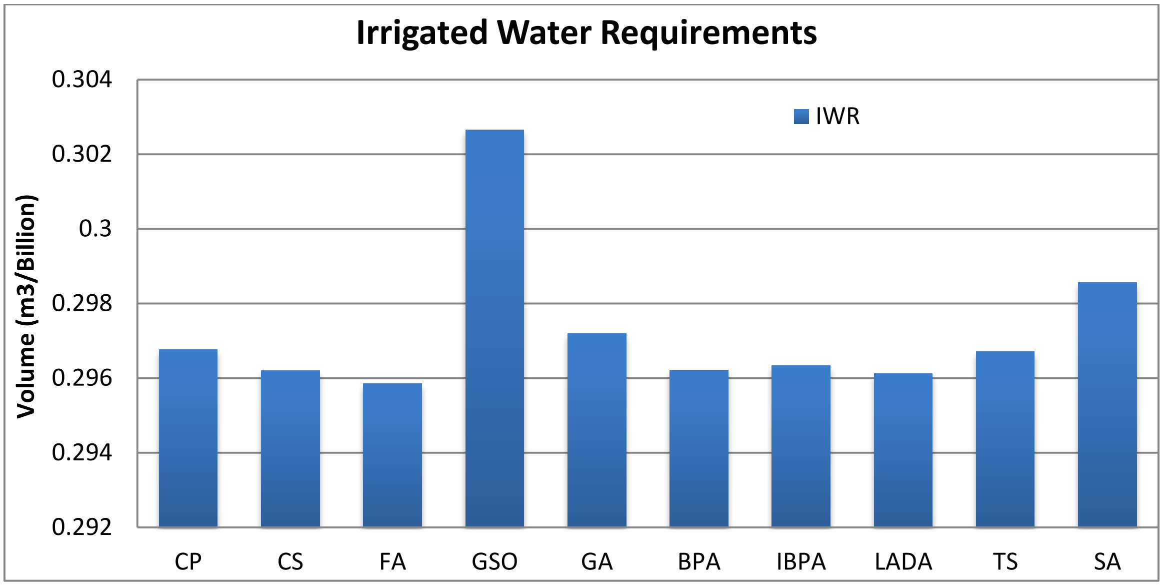

Figure A.1.5.3: Irrigated water requirements (IWR) of the best metaheuristic solutions

Figure A.1.5.3 shows that the IWR of CS, FA, BPA, IBPA and LADA are relatively lower than the CP.

A.1.5.4. Crop Hectare Allocations

Table A.1.5.4 gives the plot type hectare allocations of the best solution determined by each metaheuristic algorithm.

Table A.1.5.4: Plot type hectare allocations of each crop type

Crops	Methods								
	CS	**FA**	**GSO**	**GA**	**BPA**	**IBPA**	**LADA**	**TS**	**SA**
P/Nuts	50	65	61	115	75	66	51	98	121
W/Grapes	426	445	207	340	426	405	429	222	151
Olives	287	460	540	446	292	290	291	480	351
Lucerne	7,536	7,329	7,492	7,399	7,507	7,540	7,529	7,500	7,677
Cotton	1,330	1,324	1,338	1,299	1,333	1,332	1,340	1,350	1,431
Maize	7,915	7,870	7,908	7,659	7,909	7,914	7,906	7,848	7,667
G/Nuts	5,984	5,962	5,950	5,837	5,981	5,980	5,976	5,966	5,814
Tomato	3,857	3,828	3,832	3,707	3,852	3,853	3,847	3,821	3,616
Pumpkin	67	153	67	107	71	71	67	147	165
S/Flower	798	796	794	1,192	801	798	805	801	940
D/Beans	133	150	174	275	136	134	134	140	219
S/Beans	67	67	85	74	68	68	74	77	298
Barley	1,582	1,601	1,803	2,076	1,584	1,594	1,605	1,658	2,181
Wheat	10,547	10,597	12,001	10,901	10,553	10,550	10,544	10,531	10,983
Onion	2,004	2,006	1,803	1,760	2,002	2,002	2,001	1,970	1,510
Potato	2,426	2,418	1,210	2,188	2,426	2,425	2,420	2,412	1,879
Cabbage	475	413	241	329	471	474	470	460	445
W/Melons	369	435	437	544	370	370	369	405	487
C/Flower	527	458	504	368	524	517	526	503	448
Lettuce	1,371	1,371	1,302	1,133	1,369	1,368	1,365	1,359	1,367

A graphical representation of the statistical values given in Table A.1.5.4 is given in Figure A.1.5.4 below.

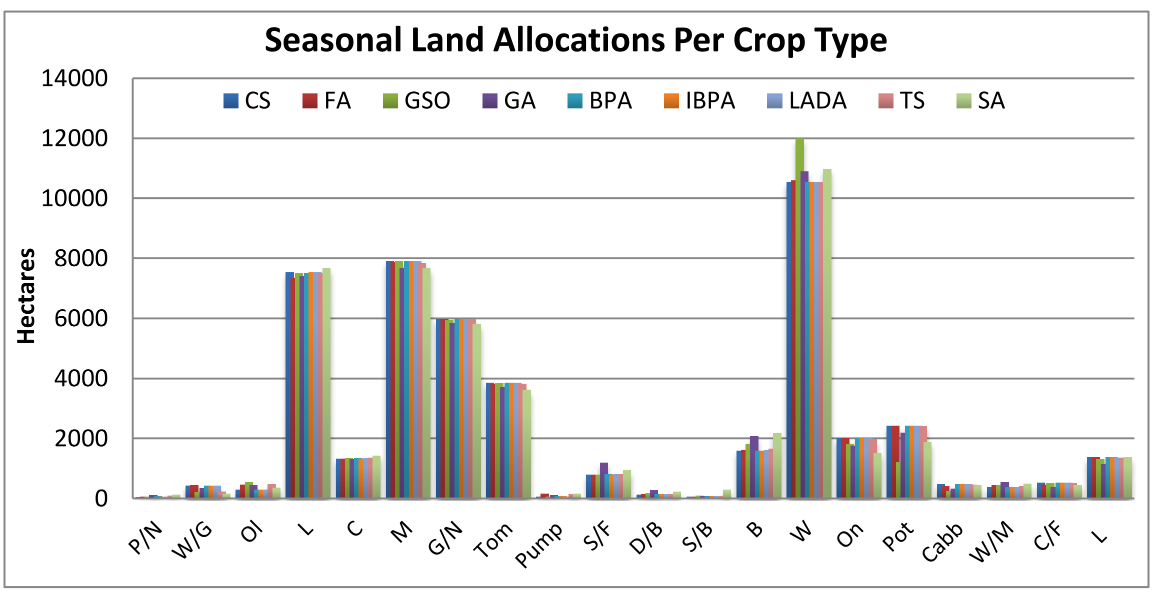

Figure A.1.5.4: A comparison of the hectare allocations, per crop, for the best solution found by each metaheuristic algorithm

A.1.6. Test Dataset 6

This dataset also consists of 20 crops. It is an extension of test dataset 4 with the addition of the 5 crops included in test dataset 5. The hectare allocations of each of the additional 5 crops in this dataset has been set differently compared to the same crops listed in test dataset 5. The total area of land allocated for the Perennial crops is 7,300 ha, the total area of land allocated for the summer crops is 21,100 ha, and the total area of land allocated for the winter crops is 26,300 ha.

Table A.6: Test dataset 6

Crop Types	**Crop**	**ha's crop^{-1}**	**t ha^{-1}**	**CWR**	**AR**	**Lower Bound**	**Upper Bound**	**ZAR t^{-1}**
Perennial	Pecan Nuts	1,000	5.0	1,600	444.7	500	1,500	3,500.0
	Wine Grapes	2,300	9.5	850	350.8	1,500	3,500	2,010.00
	Olives	2,500	6.0	1,200	444.7	1,800	3,800	2,500.00
	Lucerne	1,500	16.0	1,445	444.7	500	3,000	1,185.52
Summer	Cotton	500	3.5	700	386.4	250	800	4,500.00
	Maize	9,500	9.0	979	279.0	7,000	12,000	1,321.25
	Groundnuts	1,500	3.0	912	339.5	1,000	3,000	5,076.00
	Tomato	500	50.0	1,132	350.8	250	800	4,332.00
	Pumpkin	1,200	20.0	794	279.0	450	2,000	1,577.09
	Sunflower	4,200	3.0	648	314.9	3,200	5,800	3,739.00
	Dry Beans	1,200	2.0	650	269.2	600	2,200	5,600.00
	Soya Beans	2,500	3.0	600	269.2	1,500	4,500	2,528.01
Winter	Barley	7,200	6.0	530	58.3	5,800	9,500	2,083.27
	Wheat	2,000	6.0	650	58.3	1,200	3,000	2,174.64
	Onion	3,400	30.0	429	177.0	2,300	4,500	2,397.90
	Potato	2,700	28.0	365	152.8	1,900	3,800	2,463.00
	Cabbage	3,300	50.0	350	152.8	2,500	5,000	1,437.58
	Water Melon	3,500	20.0	500	22.4	2,500	5,000	934.00
	Cauliflower	1,400	10.0	500	152.8	600	2,500	4,252.00
	lettuce	2,800	20.0	300	33.7	1,500	4,000	4,432.00

A.1.6.1. Average Execution Times

The average execution times of the algorithms for test dataset 6 are given in Table A.1.6.1.

Table A.1.6.1: The average execution times, in milliseconds, and the 95% confidence interval values of each metaheuristic algorithm

Methods	AVG (ms)	95% CI
CS	1,673	AVG ± 13.1
FA	4,053	AVG ± 16.1
GSO	803	AVG ± 12.9
GA	1,032	AVG ± 8.1
BPA	339	AVG ± 3.7
IBPA	319	AVG ± 3.5
LADA	198	AVG ± 4.2
TS	322	AVG ± 5.3
SA	266	AVG ± 5.9

A graphical representation of the statistical values given in Table A.1.6.1 is shown in Figure A.1.6.1 below.

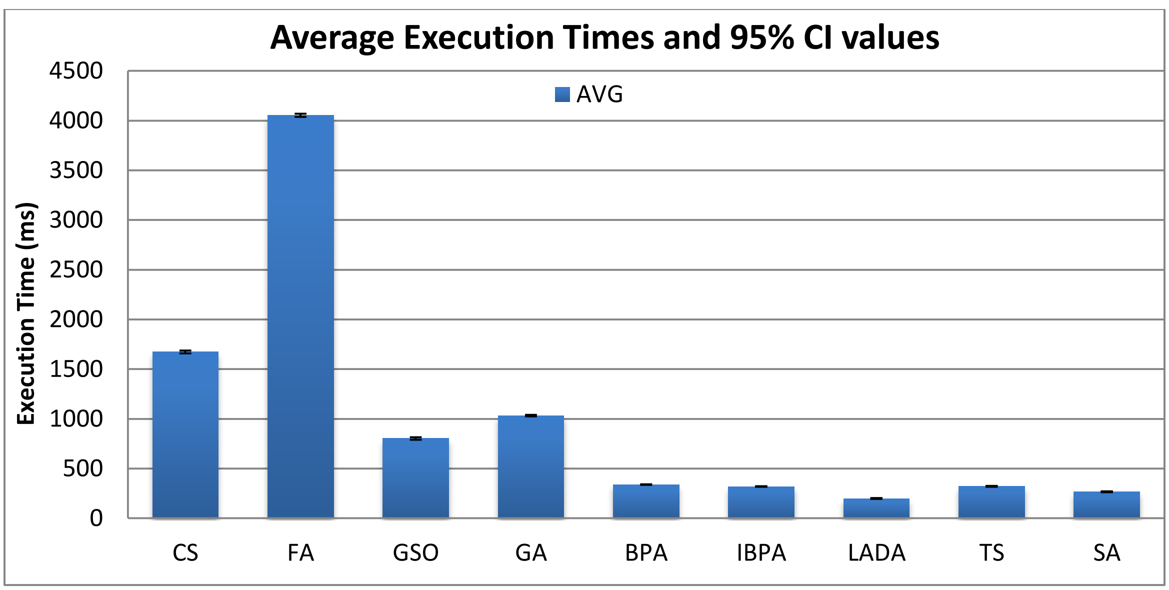

Figure A.1.6.1: The average execution times, in milliseconds (ms), and the 95% CI values of each metaheuristic algorithm

A.1.6.2. Best and Average Fitness Values

Table A.1.6.2 gives the statistical values of the BFV and ABFV values of each metaheuristic algorithm. It also gives the BFV of the CP and the 95% CI fitness values.

Table A.1.6.2: Statistics for the best fitness values (BFV), average best fitness values (ABFV) and 95% confidence interval (95% CI) values

Methods	BFV (ZAR)	ABFV (ZAR)	95% CI
CP	1,079,260,957	N/A	N/A
CS	1,260,171,850	1,260,107,403	ABFV ± 6,452
FA	1,251,293,827	1,229,429,484	ABFV ± 2,148,021
GSO	1,194,524,453	1,104,380,831	ABFV ± 6,093,006
GA	1,201,341,963	1,184,142,667	ABFV ± 1,372,857
BPA	1,258,252,113	1,257,153,897	ABFV ± 103,616
IBPA	1,258,799,974	1,257,533,739	ABFV ± 93,407
LADA	1,258,386,709	1,257,564,089	ABFV ± 66,794
TS	1,245,701,248	1,242,221,913	ABFV ± 335,751
SA	1,193,462,905	1,174,651,260	ABFV ± 1,367,235

Again CS, IBPA, BPA and LADA were the best metaheuristic algorithms. LADA however performed better than BPA and had a better ABFV than both IBPA and BPA. CS was again the best overall. IBPA was the best LS metaheuristic algorithm. A graphical comparison of the statistical values given in Table A.1.6.2 is shown in Figure A.1.6.2 below.

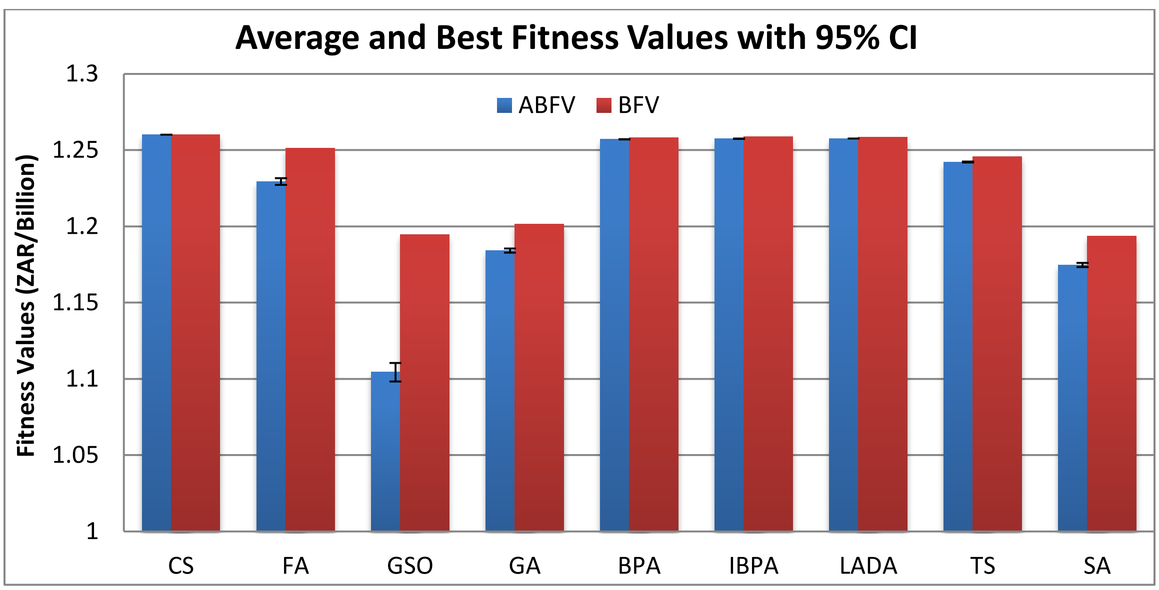

Figure A.1.6.2: A comparison of each algorithms best and average fitness values determined, along with the 95% CI estimates

A.1.6.3. Irrigated Water Requirements

Table A.1.6.3 gives the IWR's of each metaheuristic algorithm and that of the CP.

Table A.1.6.3: Statistics of the irrigation water requirements (IWR)

Methods	IWR (m^3)
CP	266,127,500
BPA	258,268,634
IBPA	256,555,066
LADA	256,289,353
TS	262,469,623
SA	259,788,342
CS	259,352,209
FA	258,435,614
GSO	259,832,205
GA	258,993,284

A graphical representation of the statistics given in Table A.1.6.3 is shown in Figure A.1.6.3 below.

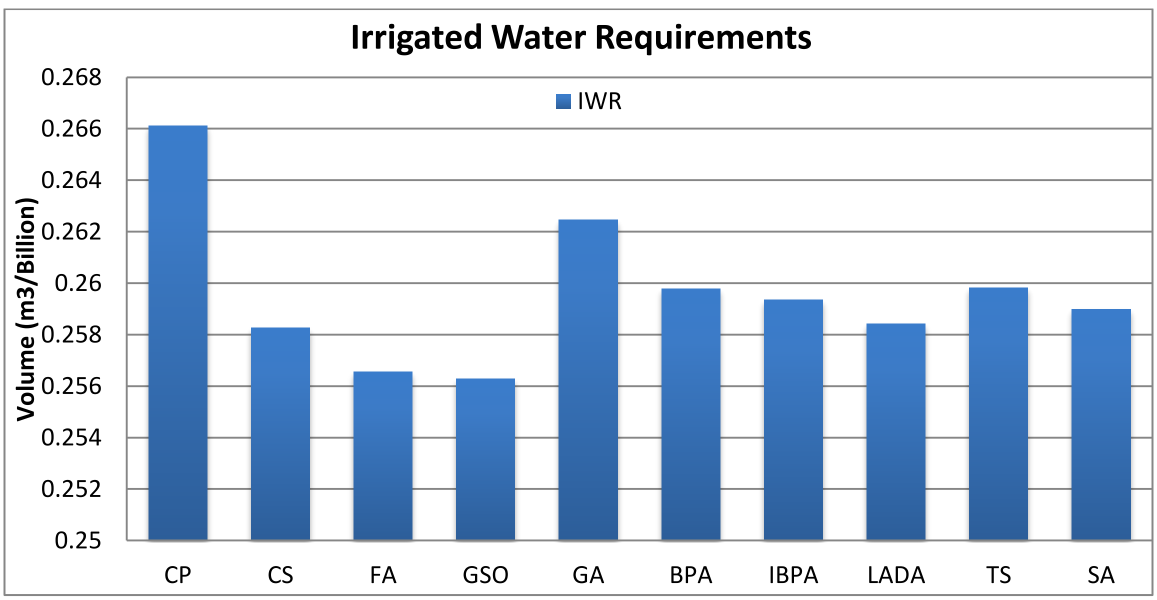

Figure A.1.6.3: Irrigated water requirements (IWR) of the best metaheuristic solutions

Figure A.1.6.3 shows that the IWR values of all metaheuristic algorithms are better than the IWR of the CP.

A.1.6.4. Crop Hectare Allocations

Table A.1.6.4 gives the plot type hectare allocations.

Table A.1.6.4: Plot type hectare allocations of each crop type

Crops	Methods								
	CS	FA	GSO	GA	BPA	IBPA	LADA	TS	SA
P/Nuts	506	783	762	1,210	857	747	521	614	748
W/Grapes	2,645	2,485	1,614	1,764	2,473	2,532	2,614	1,892	2,205
Olives	1,956	3,046	3,805	2,404	1,907	1,902	1,975	2,711	2,858
Lucerne	2,193	986	1,119	1,922	2,064	2,118	2,190	2,082	1,489
Cotton	341	372	301	499	342	355	347	495	348
Maize	9,547	9,348	8,634	9,595	9,501	9,537	9,552	9,362	8,964
G/Nuts	1,364	1,694	1,959	1,567	1,423	1,382	1,364	1,460	2,471
Tomato	777	752	674	680	773	775	775	757	587
Pumpkin	1,841	1,698	621	587	1,830	1,831	1,819	1,538	1,265
S/Flower	4,364	4,274	4,991	5,374	4,345	4,360	4,371	4,374	3,913
D/Beans	818	803	2,105	819	839	817	822	1,081	914
S/Beans	2,046	2,158	1,816	1,978	2,046	2,044	2,049	2,033	2,638
Barley	6,081	6,073	6,206	6,146	6,087	6,089	6,087	6,137	6,754
Wheat	1,235	1,235	1,269	1,256	1,238	1,238	1,236	1,251	1,215
Onion	4,115	4,096	4,044	3,796	4,112	4,104	4,111	4,097	4,001
Potato	3,475	3,465	3,520	3,421	3,468	3,472	3,472	3,295	3,254
Cabbage	4,572	4,552	4,605	3,671	4,564	4,573	4,557	4,576	4,369
W/Melons	2,561	2,561	2,596	2,555	2,565	2,567	2,580	2,655	2,561
C/Flower	604	669	1,306	1,830	616	606	606	616	758
Lettuce	3,658	3,648	2,753	3,625	3,648	3,651	3,652	3,674	3,390

A graphical representation of the statistical values given in Table A.1.6.4 is shown in Figure A.1.6.4 below.

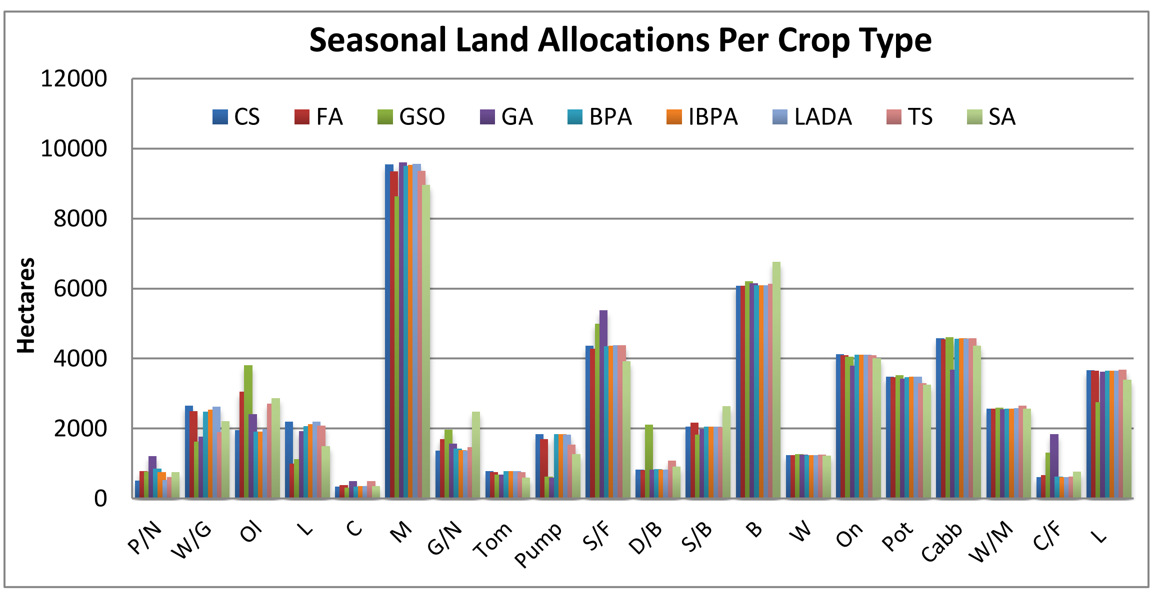

Figure A.1.6.4: A comparison of the hectare allocations, per crop, for the best solution found by each metaheuristic algorithm

A.2. New Irrigation Scheme

In contrast to an existing irrigation scheme, the optimized hectare allocations for the crops at a new irrigation scheme would need to be determined. This includes the land allocations of the different farm plot types. The solutions determined for this problem would suggest to the crop planners what would be the ideal hectare allocations of the farm plot types in progressing forward.

For the test datasets at new irrigation schemes, the crop types, the crop names, the tons of yield per hectare (t ha^{-1}), the Crop Water Requirement (CWR), the average rainfall (AR), the lower and upper bounds and the producer prices per ton of yield (ZAR t^{-1}) of each crop is given. Similar to the test datasets of an existing irrigation scheme, two datasets consist of 12 crops, two consist of 15 crops and two consist of 20 crops.

The 6 test datasets given below are the same as those found in test datasets 1 to 6. The exception is the exclusion of the hectare allocations of each crop type. However, for these datasets, the total area of land available for agricultural production at a new irrigation scheme is specified.

A.2.1. Test Dataset 7

The total area of agricultural land available for crop production for this dataset is 10,000 ha's.

Table A.7: Test dataset 7

Crop Types	Crop	t ha^{-1}	CWR	AR	Lower Bound	Upper Bound	ZAR t^{-1}
Perennial	Pecan Nuts	5.0	1,600	444.7	10	9,890	3,500.0
	Lucerne	16.0	1,445	444.7	10	9,890	1,185.52
Summer	Cotton	3.5	700	386.4	10	9,890	4,500.00
	Maize	9.0	979	279.0	10	9,890	1,321.25
	Groundnuts	3.0	912	339.5	10	9,890	5,076.00
	Tomato	50.0	1,132	350.8	10	9,890	4,332.00
	Pumpkin	20.0	794	279.0	10	9,890	1,577.09
Winter	Barley	6.0	530	58.3	10	9,890	2,083.27
	Wheat	6.0	650	58.3	10	9,890	2,174.64
	Onion	30.0	429	177.0	10	9,890	2,397.90
	Potato	28.0	365	152.8	10	9,890	2,463.00
	Cabbage	50.0	350	152.8	10	9,890	1,437.58

A.2.1.1. Average Execution Times

The average execution times of the algorithms in determining solutions for test dataset 7 are given in Table A.2.1.1. Table A.2.1.1 gives the statistics of the average execution times (AVG) in milliseconds (ms), and the 95% Confidence Interval (95% CI) values.

Table A.2.1.1: The Average execution times, in milliseconds, and the 95% confidence interval values of each metaheuristic algorithm

Methods	AVG (ms)	95% CI
CS	1,047	AVG ± 28.9
FA	4,293	AVG ± 130.9
GSO	923	AVG ± 33.7
GA	1,033	AVG ± 27.5
BPA	282	AVG ± 23.1
IBPA	239	AVG ± 5.8
LADA	206	AVG ± 9.7
TS	223	AVG ± 7.9
SA	244	AVG ± 7.1

As can be observed from Table A.2.1.1, LADAs' AVG is still the best overall. FA still shows the worst execution times. A graphical representation of the values given in Table A.2.1.1 is shown in Figure A.2.1.1 below.

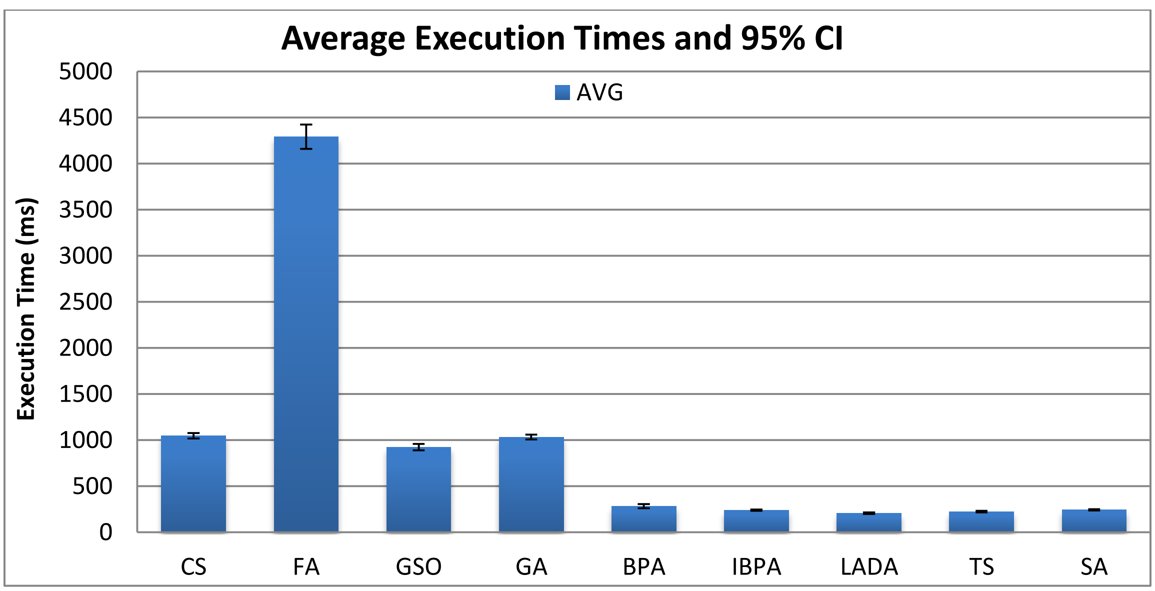

Figure A.2.1.1: The average execution times, in milliseconds (ms), and the 95% CI values of each metaheuristic algorithm

A.2.1.2. Best and Average Fitness Values

Table A.2.1.2 gives the statistical values of the overall best (BFV) and average best (ABFV) fitness values of each metaheuristic algorithm. The 95% CI values for the fitness value populations of each algorithm is also given.

Table A.2.1.2: Statistics for the best fitness values (BFV), average best fitness values (ABFV) and 95% confidence interval (95% CI) values

Methods	BFV (ZAR)	ABFV (ZAR)	95% CI
CS	1,718,276,405	1,682,327,674	ABFV ± 16,200,309
FA	1,787,089,970	1,758,445,768	ABFV ± 19,172,136
GSO	1,808,200,026	1,672,792,907	ABFV ± 118,996,166
GA	1,639,408,016	1,566,138,119	ABFV ± 32,415,994
BPA	1,763,365,587	1,713,479,214	ABFV ± 17,232,492
IBPA	1,751,148,623	1,699,990,264	ABFV ± 21,129,353
LADA	1,687,995,443	1,625,471,861	ABFV ± 22,021,582
TS	1,794,634,348	1,782,380,599	ABFV ± 4,611,740
SA	1,744,446,345	1,713,868,027	ABFV ± 13,026,862

Compared to the solutions of an existing Irrigation Scheme, the fitness value performances of the algorithms in determining solutions for a new Irrigation Scheme is more uncertain. This is due to the difficulty of determining solutions in a solution space of constantly changing dimensions. From Table A.2.1.2, it can be observed that GSO determined the best BFV and TS has the best ABFV solutions. Overall, TS was the most consistent. GA performed the worst overall. This is visually seen in Figure A.2.1.2 below.

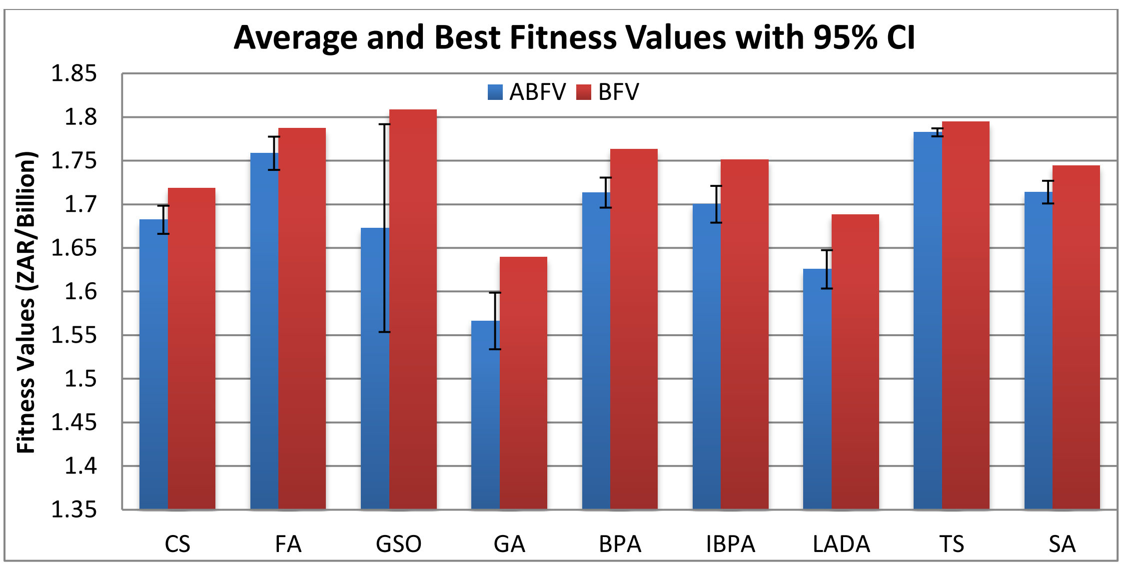

Figure A.2.1.2: A comparison of each algorithms best and average fitness values determined, along with the 95% CI estimates

A.2.1.3. Plot Type Hectare Allocations

Table A.2.1.3 gives the plot type hectare allocations for the best solution found by each metaheuristic algorithm.

Table A.2.1.3: Plot type hectare allocations for each metaheuristic algorithm

Methods	Single-Crop Plots	Double-Crop Plots
CS	598	9,402
FA	536	9,464
GSO	516	9,484
GA	778	9,222
BPA	526	9,474
IBPA	534	9,466
LADA	4,578	5,422
TS	527	9,473
SA	551	9,449

As can be observed from Table A.2.1.3, LADA allocated the most amount of land to the single-crop plots compared to the other algorithms. This is due to its stronger explorative ability. All the other algorithms performed similarly in making plot type hectare allocation decisions.

A.2.1.4. Crop Hectare Allocations

Table A.2.1.4 gives the plot type hectare allocations for each crop type, as determined by the best solution of each metaheuristic algorithm.

Table A.2.1.4: Plot type hectare allocations of each crop type

Crops	Methods								
	CS	FA	GSO	GA	BPA	IBPA	LADA	TS	SA
P/Nuts	373	68	22	491	102	108	3,881	132	334
W/Grapes	226	468	494	287	423	427	697	394	216
Cotton	95	50	20	341	119	92	200	21	89
Maize	104	30	20	226	110	22	72	21	22
G/Nuts	67	47	25	110	63	207	99	40	244
Tomato	9,069	9,306	9,400	8,452	9,109	9,066	5,028	9,338	8,960
Pumpkin	67	31	20	92	74	79	22	53	134
Barley	87	87	17	611	98	41	237	51	84
Wheat	938	25	43	84	77	184	66	54	77
Onion	1,861	3,124	7,769	3,337	3,996	1,871	2,050	2,852	3,529
Potato	1,900	2,928	48	2,365	655	2,928	1,690	1,724	153
Cabbage	4,616	3,300	1,607	2,824	4,649	4,441	1,379	4,792	5,606

A graphical representation of the statistical values given in Table A.2.1.4 is shown in Figure A.2.1.4 below.

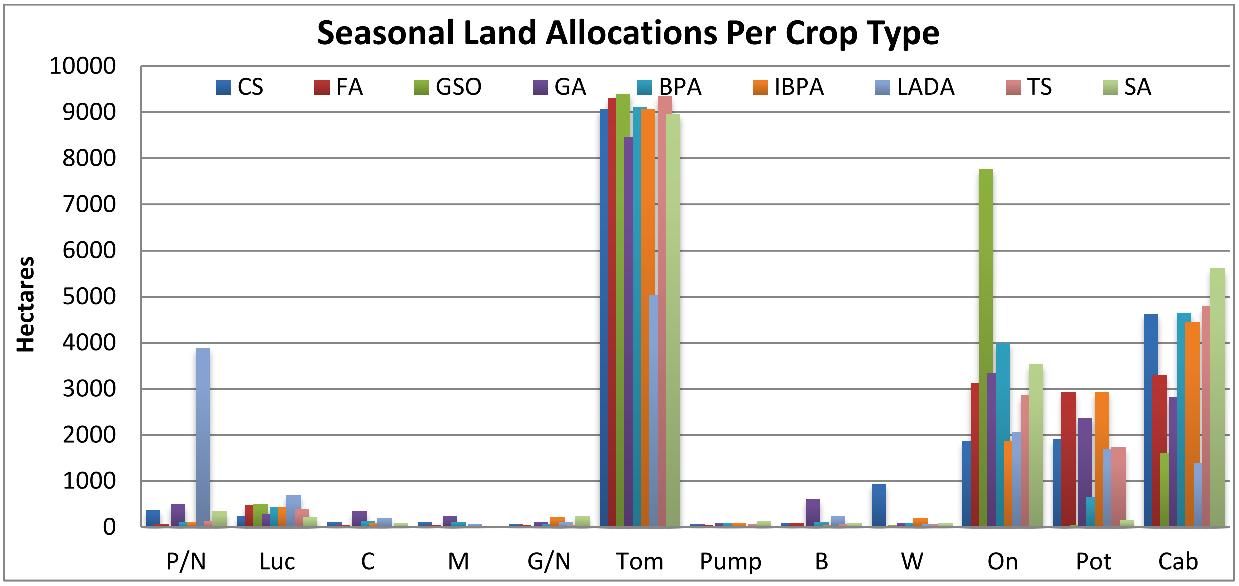

Figure A.2.1.4: A comparison of the hectare allocations, per crop, for the best solution found by each metaheuristic algorithm

As can be observed from Figure A.2.1.4, most of the algorithms determined that allocating more land to Tomato, Onion, Potato and Cabbage would determine the highest returns.

A.2.2. Test Dataset 8

The total area of agricultural land available for crop production for this dataset is 20,000 ha's.

Table A.8: Test dataset 8

Crop Types	Crop	t ha^{-1}	CWR	AR	Lower Bound	Upper Bound	ZAR t^{-1}
Perennial	Pecan Nuts	5.0	1,600	444.7	10	9,890	3,500.0
	Wine Grapes	9.5	850	350.8	10	9,890	2,010.00
	Olives	6.0	1,200	444.7	10	9,890	2,500.00
	Lucerne	16.0	1,445	444.7	10	9,890	1,185.52
Summer	Cotton	3.5	700	386.4	10	9,890	4,500.00
	Maize	9.0	979	279.0	10	9,890	1,321.25
	Groundnuts	3.0	912	339.5	10	9,890	5,076.00
	Tomato	50.0	1,132	350.8	10	9,890	4,332.00
Winter	Barley	6.0	530	58.3	10	9,890	2,083.27
	Wheat	6.0	650	58.3	10	9,890	2,174.64
	Onion	30.0	429	177.0	10	9,890	2,397.90
	Potato	28.0	365	152.8	10	9,890	2,463.00

A.2.2.1. Average Execution Times

The average execution times of the algorithms for this dataset are given in Table A.2.2.1.

Table A.2.2.1: The average execution times, in milliseconds, and the 95% confidence interval values of each metaheuristic algorithm

Methods	AVG (ms)	95% CI
CS	1,043	AVG ± 49.9
FA	4,217	AVG ± 115.5
GSO	873	AVG ± 21.8
GA	998	AVG ± 23.8
BPA	259	AVG ± 2.4
IBPA	234	AVG ± 2.7
LADA	202	AVG ± 2.6
TS	234	AVG ± 25.3
SA	241	AVG ± 3.7

As can be observed, LADAs' AVG is still the best overall. A graphical comparison of the values given in Table A.2.2.1 is shown in Figure A.2.2.1 below.

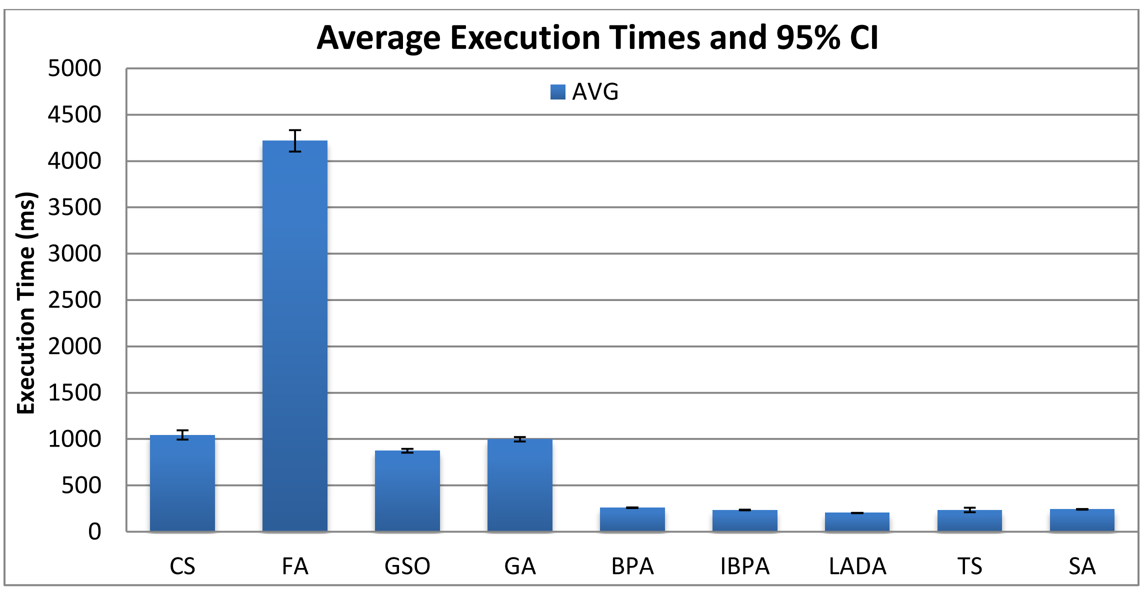

Figure A.2.2.1: The average execution times, in milliseconds (ms), and the 95% CI values of each metaheuristic algorithm

A.2.2.2. Best and Average Fitness Values

Table A.2.2.2 gives the statistical values of the BFV and ABFV values of each metaheuristic algorithm. The 95% CI fitness values is also given.

Table A.2.2.2: Statistics for the best fitness values (BFV), average best fitness values (ABFV) and 95% confidence interval (95% CI) values

Methods	BFV (ZAR)	ABFV (ZAR)	95% CI
CS	1,717,269,933	1,682,796,805	ABFV ± 18,580,922
FA	1,804,999,630	1,772,058,929	ABFV ± 18,726,435
GSO	1,791,012,803	1,505,002,640	ABFV ± 159,624,331
GA	1,672,510,247	1,565,333,506	ABFV ± 29,823,992
BPA	1,754,297,353	1,705,166,875	ABFV ± 13,100,845
IBPA	1,735,289,103	1,705,642,556	ABFV ± 13,933,737
LADA	1,696,370,221	1,654,928,771	ABFV ± 17,973,045
TS	1,799,119,889	1,785,982,013	ABFV ± 4,661,099
SA	1,782,526,402	1,707,423,528	ABFV ± 24,346,220

From Table A.2.2.2, it is observed that FA determined the best BFV and TS gave the best ABFV performance. Overall, TS again performed the most consistently. A visual representation of the values given in Table A.2.2.2 is shown in Figure A.2.2.2 below.

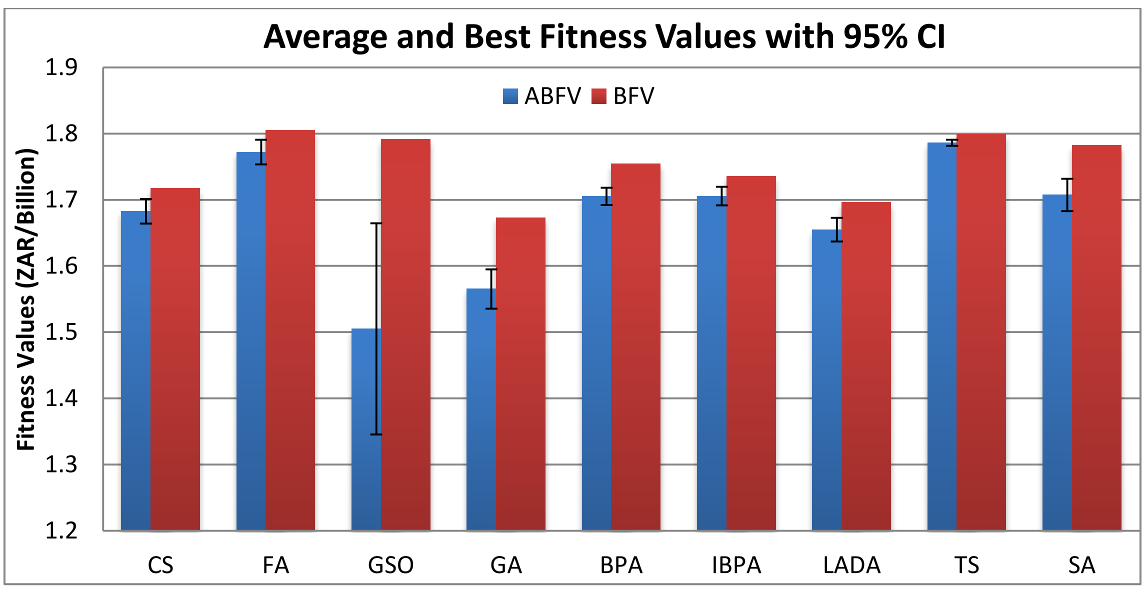

Figure A.2.2.2: A comparison of each algorithms best and average fitness values determined, along with the 95% CI estimates

A.2.2.3. Plot Type Hectare Allocations

Table A.2.2.3 gives the plot type hectare allocations of the farm plots.

Table A.2.2.3: Plot type hectare allocations for each metaheuristic algorithm

Methods	Single-Crop Plots	Double-Crop Plots
CS	561	9,439
FA	504	9,496
GSO	620	9,380
GA	625	9,375
BPA	622	9,378
IBPA	582	9,418
LADA	5,751	4,249
TS	502	9,498
SA	504	9,496

A.2.2.4. Crop Hectare Allocations

Table A.2.2.4 gives the plot type hectare allocations of each crop type as also determined by the best solution of each metaheuristic algorithm.

Table A.2.2.4: Plot type hectare allocations of each crop type

Crops	Methods								
	CS	FA	GSO	GA	BPA	IBPA	LADA	TS	SA
P/Nuts	108	212	49	76	93	211	3027	141	186
W/Grapes	453	292	571	549	529	371	2724	361	318
Cotton	62	38	19	636	82	216	36	49	37
Maize	112	33	20	36	105	77	21	39	76
G/Nuts	36	22	25	92	26	86	144	29	35
Tomato	9,196	9,378	9,295	8,557	9,081	8,918	3,964	9,360	9,305
Pumpkin	34	25	20	54	84	121	84	21	43
Barley	550	32	10	298	66	11	215	31	295
Wheat	921	30	11	202	105	202	64	114	16
Onion	1,185	4,236	4,657	3,975	4,000	4,190	1,440	4,674	4,297
Potato	2,855	336	28	516	1,290	121	742	524	1,823
Cabbage	3,928	4,862	4,674	4,384	3,917	4,894	1,788	4,155	3,065

A graphical representation of the statistics given in Table A.2.2.4 is shown in Figure A.2.2.4 below.

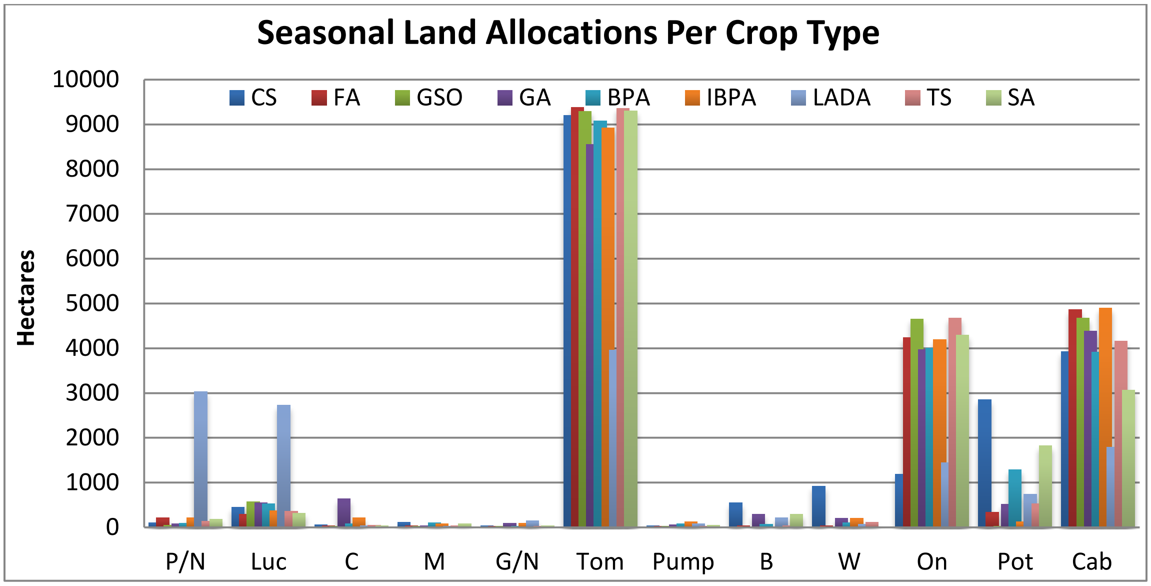

Figure A.2.2.4: A comparison of the hectare allocations, per crop, for the best solution found by each metaheuristic algorithm

A.2.3. Test Dataset 9

The total area of agricultural land allocated for this test dataset is 10,000 ha's.

Table A.9: Test dataset 9

Crop Types	Crop	t ha^{-1}	CWR	AR	Lower Bound	Upper Bound	ZAR t^{-1}
Perennial	Pecan Nuts	5.0	1,600	444.7	10	9,910	3,500.0
	Wine Grapes	9.5	850	350.8	10	9,910	2,010.00
	Olives	6.0	1,200	444.7	10	9,910	2,500.00
	Lucerne	16.0	1,445	444.7	10	9,910	1,185.52
Summer	Cotton	3.5	700	386.4	10	9,910	4,500.00
	Maize	9.0	979	279.0	10	9,910	1,321.25
	Groundnuts	3.0	912	339.5	10	9,910	5,076.00
	Tomato	50.0	1,132	350.8	10	9,910	4,332.00
	Pumpkin	20.0	794	279.0	10	9,910	1,577.09
	Sunflower	3.0	648	314.9	10	9,910	3,739.00
Winter	Barley	6.0	530	58.3	10	9,912	2,083.27
	Wheat	6.0	650	58.3	10	9,912	2,174.64
	Onion	30.0	429	177.0	10	9,912	2,397.90
	Potato	28.0	365	152.8	10	9,912	2,463.00
	Cabbage	50.0	350	152.8	10	9,912	1,437.58

A.2.3.1. Average Execution Times

The average execution times of the algorithms are given in Table A.2.3.1.

Table A.2.3.1: The average execution times, in milliseconds, and the 95% confidence interval values of each metaheuristic algorithm

Methods	AVG (ms)	95% CI
CS	1,291	AVG ± 86.2
FA	4,938	AVG ± 141.1
GSO	950	AVG ± 16.9
GA	1,097	AVG ± 42.2
BPA	298	AVG ± 4.4
IBPA	276	AVG ± 4.9
LADA	228	AVG ± 6.4
TS	266	AVG ± 10.5
SA	284	AVG ± 10.3

A graphical representation of these execution times is shown in Figure A.2.3.1 below.

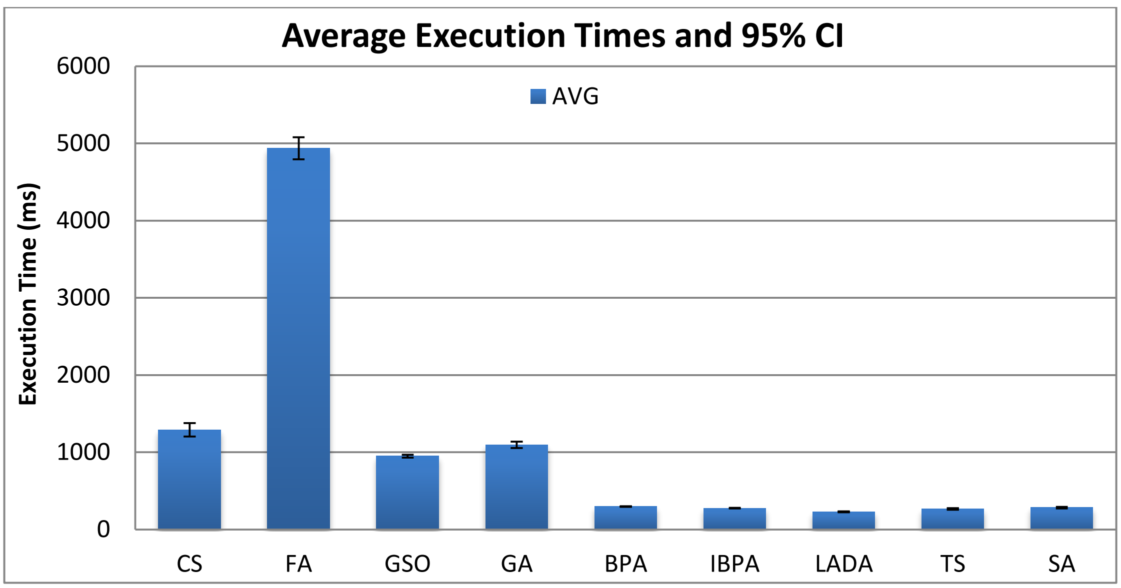

Figure A.2.3.1: The average execution times, in milliseconds (ms), and the 95% CI values of each metaheuristic algorithm

A.2.3.2. Best and Average Fitness Values

Table A.2.3.2 gives the statistical values of the BFV and ABFV values of each metaheuristic algorithm. The 95% CI fitness values are also given.

Table A.2.3.2: Statistics for the best fitness values (BFV), average best fitness values (ABFV) and 95% confidence interval (95% CI) values

Methods	BFV (ZAR)	ABFV (ZAR)	95% CI
CS	1,610,709,670	1,581,896,507	ABFV ± 21,102,679
FA	1,711,518,317	1,682,910,171	ABFV ± 21,814,803
GSO	1,645,688,301	1,174,021,690	ABFV ± 229,510,218
GA	1,481,188,526	1,434,528,695	ABFV ± 27,857,649
BPA	1,654,236,740	1,615,428,956	ABFV ± 28,482,871
IBPA	1,650,850,089	1,605,145,930	ABFV ± 27,259,032
LADA	1,637,334,948	1,528,021,131	ABFV ± 53,699,345
TS	1,713,373,531	1,701,474,497	ABFV ± 9,648,062
SA	1,682,069,214	1,641,448,908	ABFV ± 38,177,820

For this dataset, TS and FA performed the best overall. This can be seen visually in Figure A.2.3.2 below.

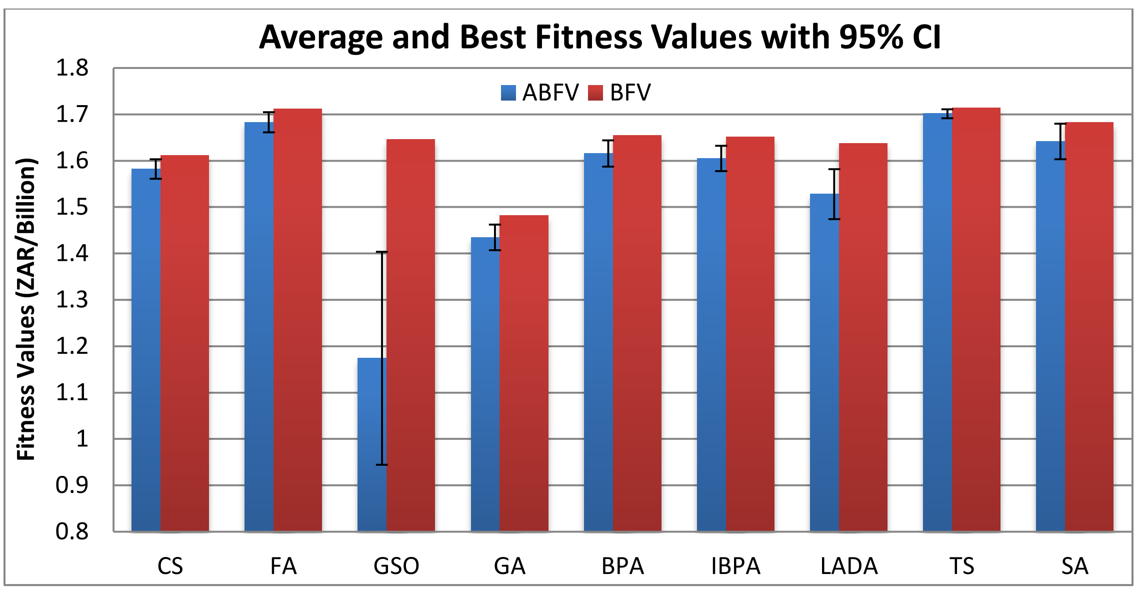

Figure A.2.3.2: A comparison of each algorithms best and average fitness values determined, along with the 95% CI estimates

A.2.3.3. Plot Type Hectare Allocations

Table A.2.3.3 gives the plot type hectare allocations for the farm plots.

Table A.2.3.3: Plot type hectare allocations for each metaheuristic algorithm

Methods	Single-Crop Plots	Double-Crop Plots
CS	636	9,364
FA	628	9,372
GSO	664	9,336
GA	743	9,257
BPA	664	9,336
IBPA	700	9,300
LADA	4,744	5,256
TS	669	9,331
SA	685	9,315

A.2.3.4. Crop Hectare Allocations

Table A.2.3.4 gives the plot type hectare allocations for each crop type as also determined by the best solution of each metaheuristic algorithm.

Table A.2.3.4: Plot type hectare allocations of each crop type

Crops	Methods								
	CS	FA	GSO	GA	BPA	IBPA	LADA	TS	SA
P/Nuts	17	200	40	19	377	165	1,301	127	155
W/Grapes	187	74	428	185	90	254	953	226	232
Olives	292	171	166	304	44	143	1587	41	73
Lucerne	140	184	29	235	154	138	904	274	225
Cotton	151	100	125	87	117	149	100	101	124
Maize	161	108	119	712	192	149	68	126	95
G/Nuts	251	123	135	327	111	291	78	101	119
Tomato	8,325	8,779	8,709	7,543	8,462	8,420	4,709	8,776	8,612
Pumpkin	223	111	124	114	238	185	215	103	181
S/flower	254	150	124	475	215	106	86	124	185
Barley	436	69	1,511	803	206	114	246	67	273
Wheat	630	81	69	767	221	242	77	77	133
Onion	2,341	3,297	2,980	1,145	1,472	4,159	1,419	3,855	3,875
Potato	4,520	2,469	1,621	2,865	4,132	3,541	1,725	659	66
Cabbage	1,437	3,456	3,155	3,677	3,304	1,245	1,789	4,673	4,968

A graphical representation of the statistics given in Table A.2.3.4 is shown in Figure A.2.3.4 below.

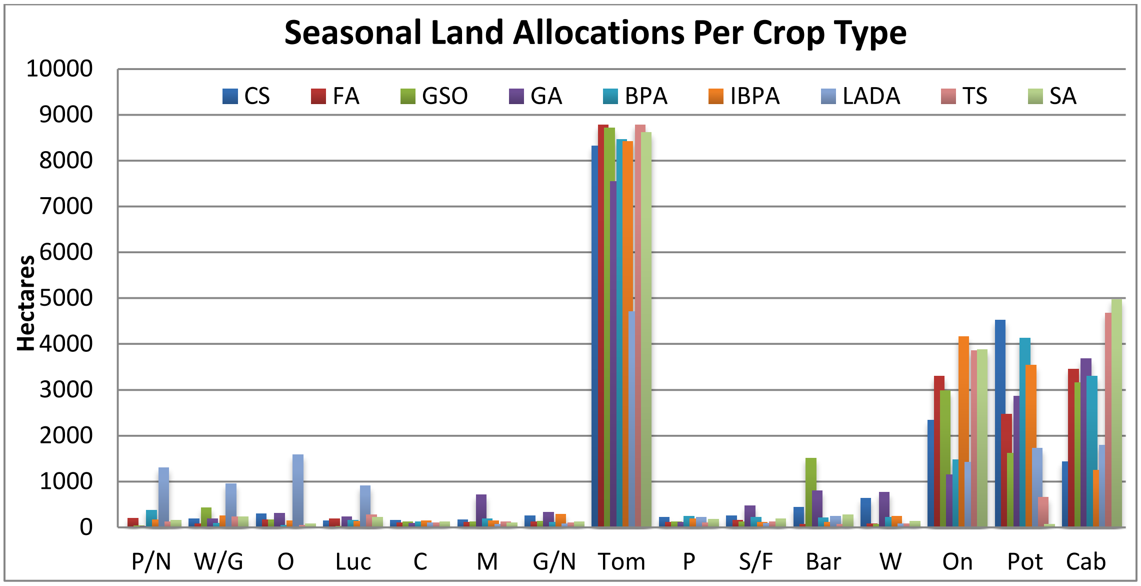

Figure A.2.3.4: A comparison of the hectare allocations, per crop, for the best solution found by each metaheuristic algorithm

A.2.4. Test Dataset 10

The total area of agricultural land available for this dataset is 20,000 ha's.

Table A.10: Test dataset 10

Crop Types	Crop	t ha^{-1}	CWR	AR	Lower Bound	Upper Bound	ZAR t^{-1}
Perennial	Pecan Nuts	5.0	1,600	444.7	10	9,910	3,500.0
	Wine Grapes	9.5	850	350.8	10	9,910	2,010.00
	Olives	6.0	1,200	444.7	10	9,910	2,500.00
	Lucerne	16.0	1,445	444.7	10	9,910	1,185.52
Summer	Cotton	3.5	700	386.4	10	9,910	4,500.00
	Maize	9.0	979	279.0	10	9,910	1,321.25
	Groundnuts	3.0	912	339.5	10	9,910	5,076.00
	Tomato	50.0	1,132	350.8	10	9,910	4,332.00
	Pumpkin	20.0	794	279.0	10	9,910	1,577.09
	Sunflower	3.0	648	314.9	10	9,910	3,739.00
Winter	Barley	6.0	530	58.3	10	9,912	2,083.27
	Wheat	6.0	650	58.3	10	9,912	2,174.64
	Onion	30.0	429	177.0	10	9,912	2,397.90
	Potato	28.0	365	152.8	10	9,912	2,463.00
	Cabbage	50.0	350	152.8	10	9,912	1,437.58

A.2.4.1. Average Execution Times

The average execution times of the algorithms are given in Table A.2.4.1.

Table A.2.4.1: The average execution times, in milliseconds, and the 95% confidence interval values of each metaheuristic algorithm

Methods	AVG (ms)	95% CI
CS	1,265	AVG ± 76.9
FA	4,822	AVG ± 68.3
GSO	945	AVG ± 16.3
GA	1,082	AVG ± 25.2
BPA	295	AVG ± 2.9
IBPA	276	AVG ± 5.2
LADA	254	AVG ± 9.4
TS	259	AVG ± 3.2
SA	279	AVG ± 4.7

A graphical representation of the execution time performances is given in Figure A.2.4.1 below.

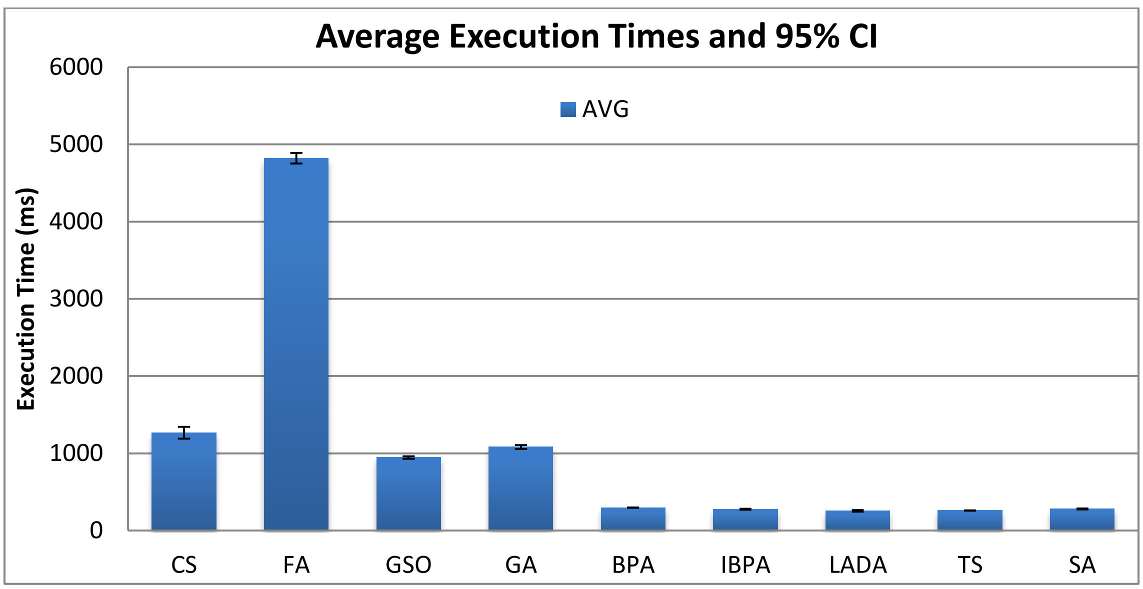

Figure A.2.4.1: The average execution times, in milliseconds (ms), and the 95% CI values of each metaheuristic algorithm

A.2.4.2. Best and Average Fitness Values

Table A.2.4.2 gives the statistical values of the BFV and ABFV values of each metaheuristic algorithm. The 95% CI fitness values is also given.

Table A.2.4.2: Statistics for the best fitness values (BFV), average best fitness values (ABFV) and 95% confidence interval (95% CI) values

Methods	BFV (ZAR)	ABFV (ZAR)	95% CI
CS	3,262,673,147	3,194,568,560	ABFV ± 34,102,233
FA	3,514,075,506	3,447,014,183	ABFV ± 50,209,803
GSO	3,557,740,772	3,228,345,873	ABFV ± 308,656,232
GA	3,012,157,550	2,884,178,098	ABFV ± 43,672,947
BPA	3,471,540,379	3,336,195,659	ABFV ± 45,186,469
IBPA	3,368,660,909	3,282,738,785	ABFV ± 33,493,865
LADA	3,218,202,294	3,085,063,455	ABFV ± 58,929,985
TS	3,530,357,906	3,504,575,225	ABFV ± 9,563,007
SA	3,469,938,794	3,358,274,990	ABFV ± 44,164,804

A graphical representation of the statistical values given in Table A.2.4.2 is shown in Figure A.2.4.2 below.

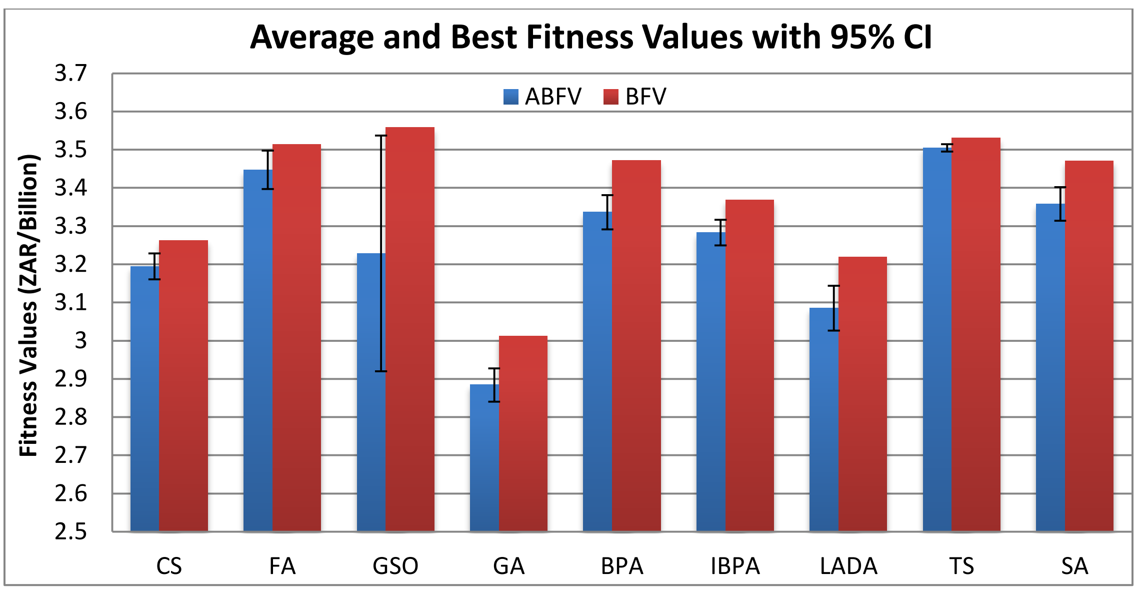

Figure A.2.4.2: A comparison of each algorithms best and average fitness values determined, along with the 95% CI estimates

A.2.4.3. Plot Type Hectare Allocations

Table A.2.4.3 gives the plot type hectare allocations for the farm plot types.

Table A.2.4.3: Plot type hectare allocations for each metaheuristic algorithm

Methods	Single-Crop Plots	Double-Crop Plots
CS	829	19,171
FA	790	19,210
GSO	671	19,329
GA	1,795	18,205
BPA	646	19,354
IBPA	639	19,361
LADA	10,978	9,022
TS	661	19,339
SA	657	19,343

A.2.4.4. Crop Hectare Allocations

Table A.2.4.4 gives the plot type hectare allocations for each crop type as also determined by the best solution of each metaheuristic algorithm.

Table A.2.4.4: Plot type hectare allocations of each crop type

Crops	Methods								
	CS	FA	GSO	GA	BPA	IBPA	LADA	TS	SA
P/Nuts	347	339	25	348	382	230	1712	51	214
W/Grapes	184	170	570	35	78	107	3446	265	296
Olives	168	135	45	721	150	60	1855	52	55
Lucerne	129	147	30	691	35	241	3965	292	92
Cotton	267	219	188	380	493	229	147	368	203
Maize	246	214	225	1,096	263	596	220	201	61
G/Nuts	676	241	191	228	410	402	197	201	316
Tomato	16,811	18,132	18,344	15,063	17,732	17,036	7,424	18,182	17,843
Pumpkin	371	204	193	680	251	785	486	194	229
S/flower	800	200	188	757	206	313	548	194	192
Barley	349	254	188	295	221	233	215	160	267
Wheat	1,881	269	188	1,450	288	567	257	422	522
Onion	8,387	8,807	15,084	6,484	8,553	7,636	3,612	8,779	7,343
Potato	7,048	4,370	2,679	2,679	2,573	4,326	931	1,605	4,806
Cabbage	1,507	5,509	1,191	7,297	7,719	6,599	4,008	8,373	6,405

A graphical representation of the statistics given in Table A.2.4.4 is shown in Figure A.2.4.4 below.

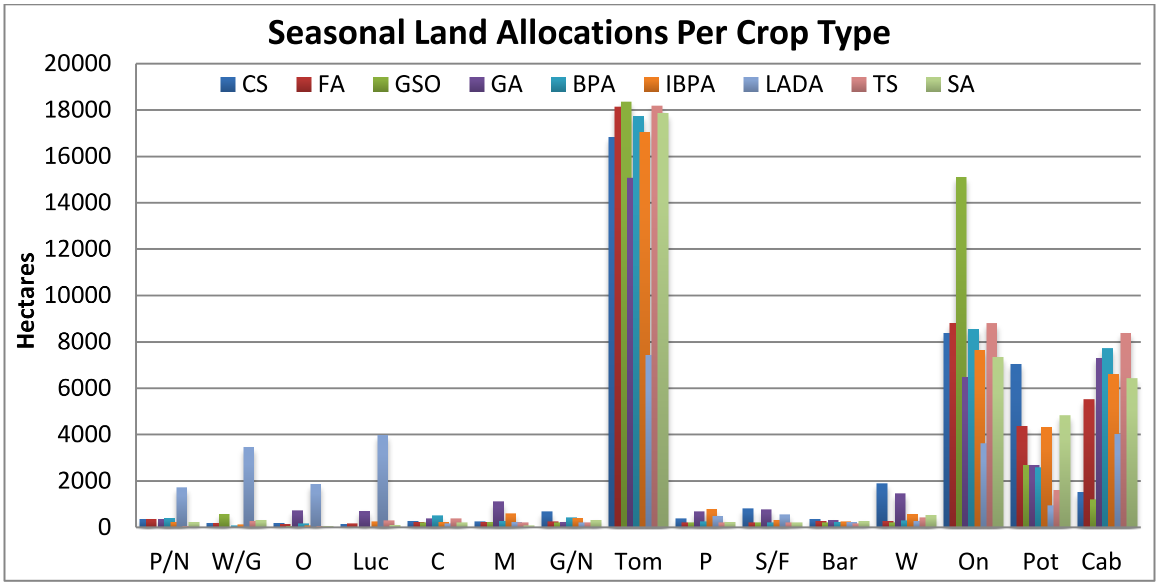

Figure A.2.4.4: A comparison of the hectare allocations, per crop, for the best solution found by each metaheuristic algorithm

A.2.5. Test Dataset 11

The total area of agricultural land available for this dataset is 10,000 ha's.

Table A.11: Test dataset 11

Crop Types	Crop	t ha-1	CWR	AR	Lower Bound	Upper Bound	ZAR t-1
Perennial	Pecan Nuts	5.0	1,600	444.7	10	9,890	3,500.0
	Wine Grapes	9.5	850	350.8	10	9,890	2,010.00
	Olives	6.0	1,200	444.7	10	9,890	2,500.00
	Lucerne	16.0	1,445	444.7	10	9,890	1,185.52
Summer	Cotton	3.5	700	386.4	10	9,890	4,500.00
	Maize	9.0	979	279.0	10	9,890	1,321.25
	Groundnuts	3.0	912	339.5	10	9,890	5,076.00
	Tomato	50.0	1,132	350.8	10	9,890	4,332.00
	Pumpkin	20.0	794	279.0	10	9,890	1,577.09
	Sunflower	3.0	648	314.9	10	9,890	3,739.00
	Dry Beans	2.0	650	269.2	10	9,890	5,600.00
	Soya Beans	3.0	600	269.2	10	9,890	2,528.01
Winter	Barley	6.0	530	58.3	10	9,890	2,083.27
	Wheat	6.0	650	58.3	10	9,890	2,174.64
	Onion	30.0	429	177.0	10	9,890	2,397.90
	Potato	28.0	365	152.8	10	9,890	2,463.00
	Cabbage	50.0	350	152.8	10	9,890	1,437.58
	Water Melon	20.0	500	22.4	10	9,890	934.00
	Cauliflower	10.0	500	152.8	10	9,890	4,252.00
	lettuce	20.0	300	33.7	10	9,890	4,432.00

A.2.5.1. Average Execution Times

The average execution times of the algorithms for this dataset are given in Table A.2.5.1.

Table A.2.5.1: The average execution times, in milliseconds, and the 95% confidence interval values of each metaheuristic algorithm

Methods	AVG (ms)	95% CI
CS	1,626	AVG ± 32.5
FA	6,069	AVG ± 68.1
GSO	1,118	AVG ± 34.5
GA	1,270	AVG ± 29.1
BPA	373	AVG ± 4.8
IBPA	346	AVG ± 3.2
LADA	279	AVG ± 3.1
TS	332	AVG ± 2.1
SA	350	AVG ± 4.1

A graphical representation of the execution time performances of the algorithms is given in Figure A.2.5.1 below.

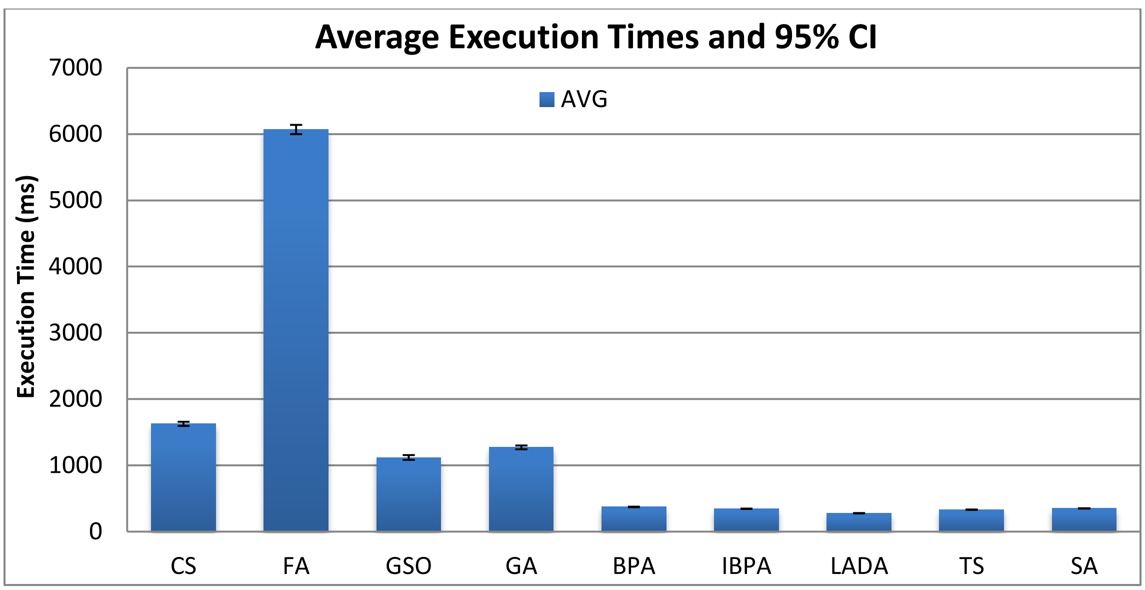

Figure A.2.5.1: The average execution times, in milliseconds (ms), and the 95% CI values of each metaheuristic algorithm

A.2.5.2. Best and Average Fitness Values

Table A.2.5.2 gives the statistical values of the BFV and ABFV values of each metaheuristic algorithm. The 95% CI fitness values is also given.

Table A.2.5.2: Statistics for the best fitness values (BFV), average best fitness values (ABFV) and 95% confidence interval (95% CI) values

Methods	BFV (ZAR)	ABFV (ZAR)	95% CI
CS	1,468,231,312	1,413,908190	ABFV ± 27,621,277
FA	1,731,781,471	1,631,522,354	ABFV ± 33,305,453
GSO	1,221,123,781	898,823,539	ABFV ± 139,268,092
GA	1,231,409,024	1,140,485,224	ABFV ± 32,362,504
BPA	1,652,258,800	1,541,708,943	ABFV ± 35,258,372
IBPA	1,594,108,853	1,521,369,602	ABFV ± 24,002,645
LADA	1,421,119,721	1,344,960,509	ABFV ± 25,879,300
TS	1,626,096,701	1,603,835,824	ABFV ± 9,861,072
SA	1,593,957,846	1,538,132,311	ABFV ± 27,043,105

For this dataset, FA performed the best overall. This was followed by BPA. A visual representation of the fitness value performances is given in Figure A.2.5.2 below.

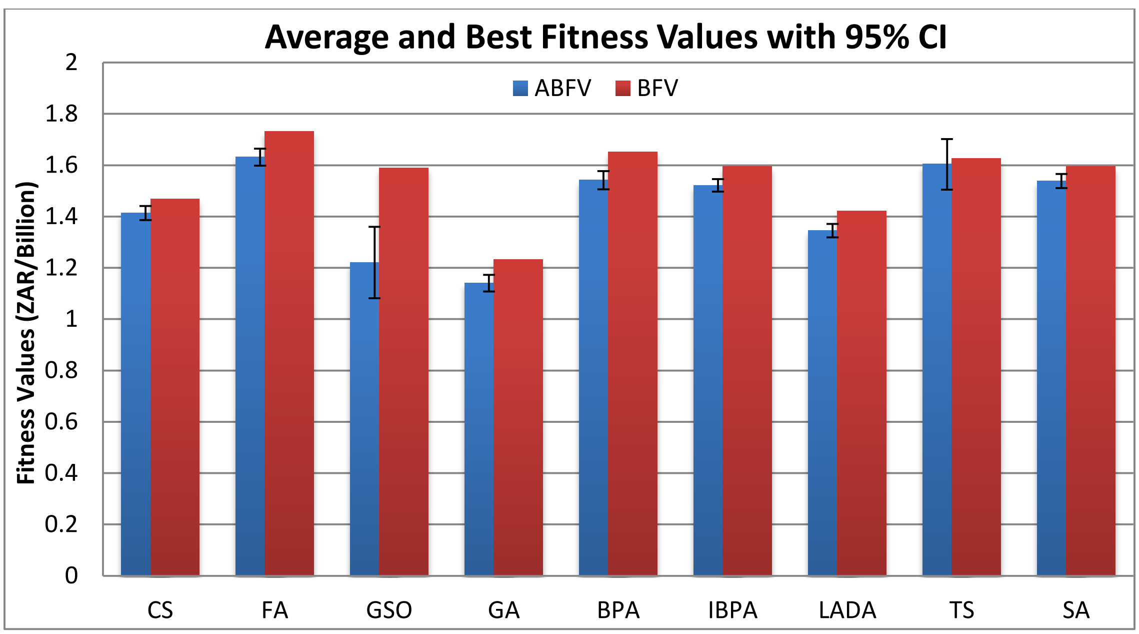

Figure A.2.5.2: A comparison of each algorithms best and average fitness values determined, along with the 95% CI estimates

A.2.5.3. Plot Type Hectare Allocations

Table A.2.5.3 gives the plot type hectare allocations for the farm plots.

Table A.2.5.3: Plot type hectare allocations for each metaheuristic algorithm

Methods	Single-Crop Plots	Double-Crop Plots
CS	806	9,194
FA	630	9,370
GSO	598	9,402
GA	601	9,399
BPA	594	9,406
IBPA	711	9,289
LADA	1,526	8,474
TS	797	9,203
SA	790	9,210

A.2.5.4. Crop Hectare Allocations

Table A.2.5.4 gives the plot type hectare allocations of each crop type as also determined by the best solution of each algorithm.

Table A.2.5.4: Plot type hectare allocations of each crop type

Crops	Methods								
	CS	FA	GSO	GA	BPA	IBPA	LADA	TS	SA
P/Nuts	150	125	167	99	124	53	193	179	241
W/Grapes	226	204	272	199	136	281	405	241	367
Olives	185	18	146	113	37	187	184	70	135
Lucerne	245	283	14	190	298	190	744	306	47
Cotton	237	121	285	76	156	132	193	88	348
Maize	210	100	215	1,192	91	634	231	322	102
G/Nuts	382	124	2,913	594	191	154	203	87	161
Tomato	7,582	8,545	5,155	5,696	8,204	7,902	6,102	8,175	7,805
Pumpkin	217	106	59	384	254	129	542	133	218
S/flower	87	132	81	434	148	113	128	107	91
D/Beans	109	114	577	397	244	96	488	153	270
S/Beans	370	128	116	627	117	131	587	138	215
Barley	1,351	130	139	187	357	312	513	430	301
Wheat	202	191	140	1,096	51	254	267	106	186
Onion	2,411	432	78	1,017	3,039	507	1,787	863	648
Potato	2,367	592	2,729	1,918	524	1034	1,640	1,840	2,149
Cabbage	1,996	1,951	2,288	1,608	1,884	3,372	1,657	1,652	175
W/Melon	216	116	429	686	344	84	292	105	156
C/Flower	575	222	733	903	49	623	177	802	757
Lettuce	77	5,736	2,867	1,986	3,157	3,104	2,142	3,403	4,837

A graphical representation of the statistics given in Table A.2.5.4 is shown in Figure A.2.5.4 below.

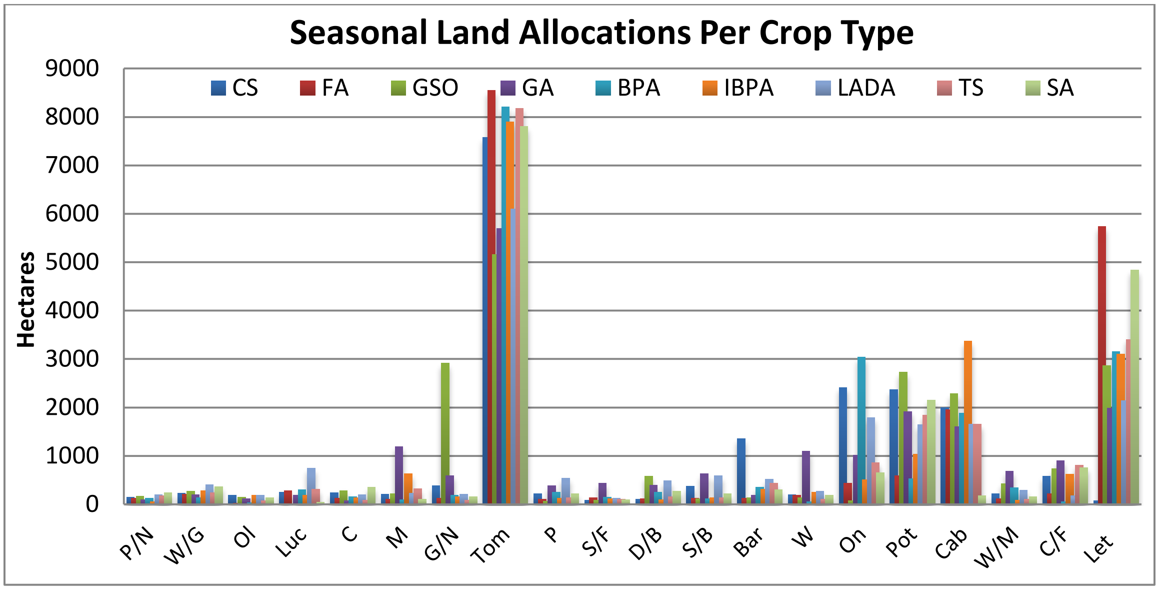

Figure A.2.5.4: A comparison of the hectare allocations, per crop, for the best solution found by each metaheuristic algorithm

A.2.6. Test Dataset 12

The total area of agricultural land available for this dataset is 20,000 ha's.

Table A.12: Test dataset 12

Crop Types	Crop	t ha⁻¹	CWR	AR	Lower Bound	Upper Bound	ZAR t⁻¹
Perennial	Pecan Nuts	5.0	1,600	444.7	10	9,890	3,500.0
	Wine Grapes	9.5	850	350.8	10	9,890	2,010.00
	Olives	6.0	1,200	444.7	10	9,890	2,500.00
	Lucerne	16.0	1,445	444.7	10	9,890	1,185.52
Summer	Cotton	3.5	700	386.4	10	9,890	4,500.00
	Maize	9.0	979	279.0	10	9,890	1,321.25
	Groundnuts	3.0	912	339.5	10	9,890	5,076.00
	Tomato	50.0	1,132	350.8	10	9,890	4,332.00
	Pumpkin	20.0	794	279.0	10	9,890	1,577.09
	Sunflower	3.0	648	314.9	10	9,890	3,739.00
	Dry Beans	2.0	650	269.2	10	9,890	5,600.00
	Soya Beans	3.0	600	269.2	10	9,890	2,528.01
Winter	Barley	6.0	530	58.3	10	9,890	2,083.27
	Wheat	6.0	650	58.3	10	9,890	2,174.64
	Onion	30.0	429	177.0	10	9,890	2,397.90
	Potato	28.0	365	152.8	10	9,890	2,463.00
	Cabbage	50.0	350	152.8	10	9,890	1,437.58
	Water Melon	20.0	500	22.4	10	9,890	934.00
	Cauliflower	10.0	500	152.8	10	9,890	4,252.00
	lettuce	20.0	300	33.7	10	9,890	4,432.00

A.2.6.1. Average Execution Times

The average execution times of the algorithms for this dataset are given in Table A.2.6.1.

Table A.2.6.1: The average execution times, in milliseconds, and the 95% confidence interval values of each metaheuristic algorithm

Methods	AVG (ms)	95% CI
CS	1,603	AVG ± 38.9
FA	5,991	AVG ± 88.0
GSO	1,074	AVG ± 16.6
GA	1,257	AVG ± 27.7
BPA	367	AVG ± 2.8
IBPA	339	AVG ± 3.2
LADA	276	AVG ± 3.4
TS	326	AVG ± 2.2
SA	346	AVG ± 7.1

A graphical representation of the execution time performances is given in Figure A.2.6.1 below.

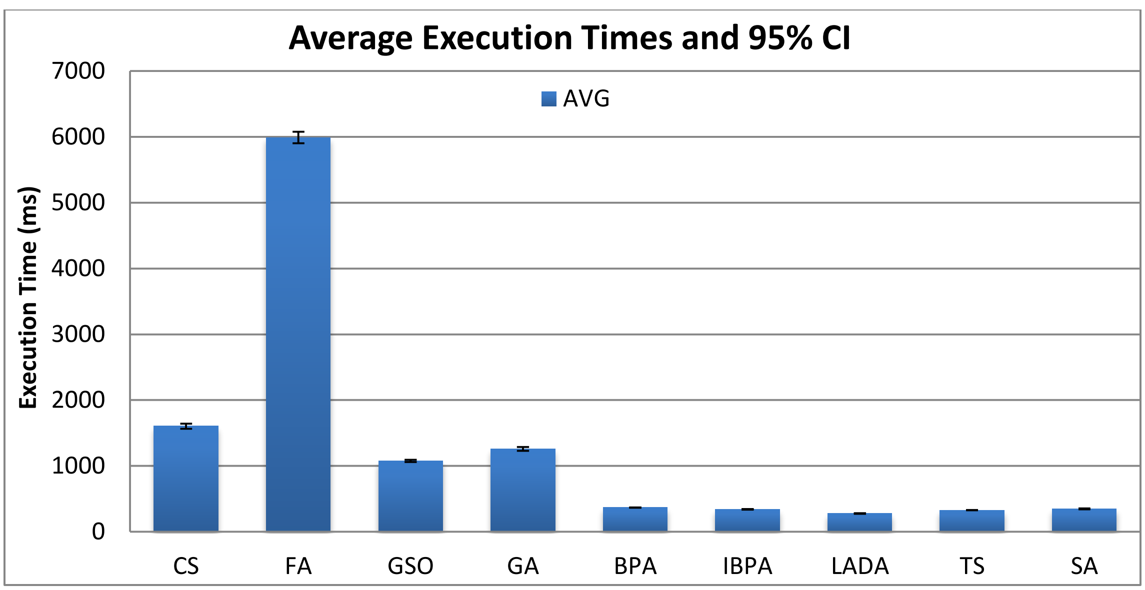

Figure A.2.6.1: The average execution times, in milliseconds (ms), and the 95% CI values of each metaheuristic algorithm

A.2.6.2. Best and Average Fitness Values

Table A.2.6.2 gives the statistical values of the BFV and ABFV values of each metaheuristic algorithm. The 95% CI fitness values is also given.

Table A.2.6.2: Statistics for the best fitness values (BFV), average best fitness values (ABFV) and 95% confidence interval (95% CI) values

Methods	BFV (ZAR)	ABFV (ZAR)	95% CI
CS	2,920,041,406	2,825,297,678	ABFV ± 52,044,585
FA	3,428,234,563	3,260,593,213	ABFV ± 63,914,114
GSO	2,740,310,583	1,714,495,779	ABFV ± 230,564,859
GA	2,612,756,569	2,297,555,653	ABFV ± 90,445,429
BPA	3,223,706,055	3,122,522,187	ABFV ± 41,966,964
IBPA	3,217,308,789	3,061,306,477	ABFV ± 53,539,509
LADA	2,804,236,704	2,661,080,953	ABFV ± 41,490,755
TS	3,313,893,985	3,249,081,678	ABFV ± 27,823,967
SA	3,181,663,979	3,078,341,017	ABFV ± 45,387,969

From Table A.2.6.2, it can be observed that FA performed the best. This was followed by TS, BPA and IBPA. The fitness value performances can be seen in Figure A.2.6.2 below.

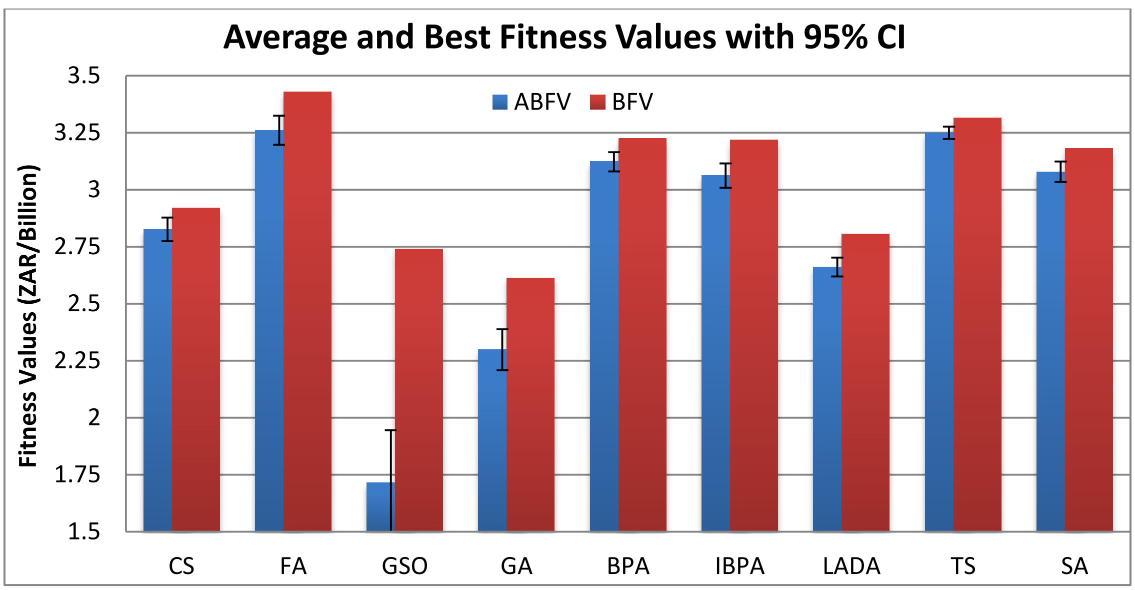

Figure A.2.6.2: A comparison of each algorithms best and average fitness values determined, along with the 95% CI estimates

A.2.6.3. Plot Type Hectare Allocations

Table A.2.6.3 gives the plot type hectare allocations for the farm plots.

Table A.2.6.3: Plot type hectare allocations for each metaheuristic algorithm

Methods	Single-Crop Plots	Double-Crop Plots
CS	1,172	18,828
FA	1,251	18,749
GSO	1,055	18,945
GA	1,738	18,262
BPA	1,171	18,829
IBPA	1,248	18,752
LADA	8,120	11,880
TS	1,129	18,871
SA	1,196	18,804

A.2.6.4. Crop Hectare Allocations

Table A.2.6.4 gives the plot type hectare allocations for each crop type as also determined by the best solution of each metaheuristic algorithm.

Table A.2.6.4: Plot type hectare allocations of each crop type

Crops	Methods								
	CS	FA	GSO	GA	BPA	IBPA	LADA	TS	SA
P/Nuts	486	583	82	650	106	463	2,642	386	55
W/Grapes	258	340	490	218	113	294	2,649	101	698
Olives	295	69	79	687	288	57	1,518	129	280
Lucerne	133	259	405	183	664	434	1,311	512	163
Cotton	634	218	980	377	697	325	128	221	459
Maize	572	200	2,360	390	390	474	1,241	444	762
G/Nuts	1,094	259	512	583	900	491	693	200	553
Tomato	13,989	17,038	12,729	13,430	15,822	15,796	8,590	16,445	15,587
Pumpkin	458	355	1,436	358	193	282	199	272	405
S/flower	184	209	203	459	227	315	430	372	344
D/Beans	875	230	505	1,504	300	753	204	381	404
S/Beans	1,022	239	220	1,160	301	315	395	536	289
Barley	439	66	727	1,120	855	189	637	215	72
Wheat	1,089	106	179	1,975	67	120	282	208	312
Onion	2,393	4,338	9,409	1,844	2,681	3,796	2,168	5,166	5,083
Potato	5,147	5,062	5,907	5,823	3,031	5,260	2,559	4,025	4,140
Cabbage	1,347	2,117	979	708	3,734	4,625	2,221	4,427	4,606
W/Melon	354	86	414	419	674	149	685	96	478
C/Flower	1,666	166	847	5,463	324	393	985	275	191
Lettuce	6,393	6,808	482	909	7,465	4,220	2,344	4,459	3,921

A graphical representation of the statistics given in Table A.2.6.4 is shown in Figure A.2.6.4 below.

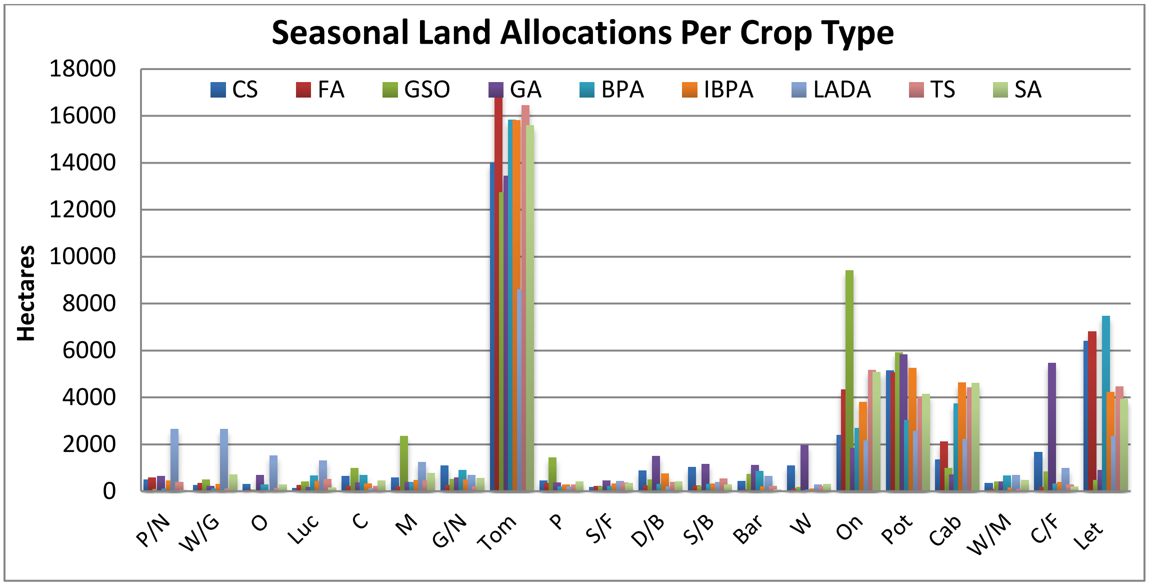

Figure A.2.6.4: A comparison of the hectare allocations, per crop, for the best solution found by each metaheuristic algorithm