UNIVERSITY OF KWAZULU NATAL
SCHOOL OF MATHEMATICS, STATISTICS AND COMPUTER SCIENCE

A Comparative Study of Metaheuristics for Blood Assignment Problem

Author: Prinolan Govender
Student Number: 213535970

Supervisor: Dr. Absalom E. Ezugwu

A THESIS SUBMITTED IN FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF MASTERS OF SCIENCE IN THE UNIVERSITY OF KWAZULU-NATAL SCHOOL OF MATHEMATICS, STATISTICS AND COMPUTER SCIENCE

I, Prinolan Govender , declare that:

(i) The research reported in this thesis, except where otherwise indicated, is my original research.

(ii) This thesis has not been submitted for any degree or examination at any other university.

(iii) This thesis does not contain other persons' data, pictures, graphs or other information, unless specifically acknowledged as being sourced from other persons.

(iv) 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, their writing has been placed inside quotation marks, and referenced.
>
> (v) 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 dissertation/thesis and in the References sections.

Candidate: Prinolan Govender

Signature: [redacted]

As the candidate's supervisor I agree to the submission of this thesis for examination.

Superviso[redacted]Ezugwu

Signature [redacted]

Abstract

Prinolan Govender
School of Mathematics, Statistics and Computer Science
Master of Computer Science

The Blood Assignment Problem (BAP) is a real world and NP-hard combinatorial optimization problem. The study of BAP is significant due to the continuous demand for blood transfusion during medical emergencies. However, the formulation of this problem faces various challenges that stretch from managing critical blood shortages, limited shelf life and, blood type incompatibility that constrain the random transfusion of blood to patients. The transfusion of incompatible blood types between patient and donor can lead to adverse side effects on the patients. Usually, the sudden need for blood units arises as a result of unforeseen trauma that requires urgent medical attention. This condition can interrupt the supply of blood units and may result in the blood bank importing additional blood products from external sources, thereby increasing its running cost and other risk factors associated with blood transfusion. This however, might have serious consequences in terms of medical emergency, running cost and supply of blood units. Therefore, by taking these factors into consideration the aforementioned study implemented five global metaheuristic optimization algorithms to solve the BAP. Each of these algorithms was hybridized with a sustainable blood assignment policy that relates to the South Africa blood banks. The objective of this study was to minimize blood product wastage with emphasis on expiry and reduction in the amount of importation from external sources. Extensive computational experiments were conducted over a total of six different datasets, and the results validate the reliability and effectiveness of each of the proposed algorithms. Results were analysed across three major aspects, namely, the average levels of importation, expiry across a finite time period and computational time experienced by each of the metaheuristic algorithms. The numerical results obtained show that the Particle Swarm Optimization algorithm was better in terms of computational time. Furthermore, none of the algorithms experienced any form of expiry within the allotted time frame. Moreover, the results also revealed that the Symbiotic Organism Search algorithm produced the lowest average result for importation; therefore, it was considered the most reliable and proficient algorithm for the BAP.

***Keywords**: Blood assignment, metaheuristics, genetic algorithm, particle swarm optimization, duellist algorithm, symbiotic organism search, grey wolf optimizer.*

List of Publications

1. Govender, P. and Ezugwu, A.E., 2018, December. A Comparative Study of Metaheuristics for Blood Assignment Problem. In Proceedings of the 2018 International Conference on Algorithms, Computing and Artificial Intelligence (p. 3). ACM.*)*.

2. Govender, P. and Ezugwu, A.E., 2019, January. A Symbiotic Organisms Search Algorithm for Blood Assignment Problem. In International Workshop on Hybrid Metaheuristics (pp. 200-208). Springer, Cham.

3. Govender, P. and Ezugwu, A.E., 2019. A Symbiotic Organisms Search Algorithm for Optimal Allocation of Blood Products. IEEE Access, 7, pp.2567-2588.

Dedication

This dissertation is dedicated to a number of individuals who have influenced my life in a positive manner. Firstly I would like to acknowledge my friends namely J. Ganesan, K.Govender, K. Naidoo, S. Shaik, N. Rambally, B. Ndaba, R.Singh, W. Govender, and A. Pather

Secondly, I would like to thank my parents and sister for supporting me during my studies and constantly providing me with encouragement and belief.

Finally, I would like to thank Dr. Absalom Ezugwu. He has become more than just a supervisor during my studies, and I regard him as a reliable friend who has believed in my capabilities, and provided me with knowledge, encouragement to perform at my maximum potential. Our constructive arguments and different views allowed me to grow as an individual, and I am truly thankful for his assistance in my projects development.

Contents

List of Figures

List of Tables

Chapter One

Introduction

1.1 Background

Human blood management is characterized by a string of factors which can prove to be complicated over time [1]. Blood banks often face the scenario of insufficient Whole Blood (WB) units in storage to allocate to patients in need. A WB unit is comprised of four main components, with each component serving a specific purpose within the blood cell [2].

- Red blood cells (RBC): Carry oxygen (O_2) throughout the body, and remove Carbon Dioxide (CO_2).
- White blood cells (WBC): Fight against biological threats within the body, and are known as the body's immune system.
- Platelets (PLT): Seal of wounds and prevent bleeding.
- Plasma: Transports nutrients and proteins to parts of the body, and holds RBC, WBC and PLT cells.

The majority of WB units in blood banks are usually attained through voluntary donations, and this in turn starts the process of blood screening to identify any potential blood related threats which could harm the receiving patient. Any infected blood donations are immediately discarded, whilst the other donations deemed as clean, are stored in ideal conditions to be used for future distribution. It is possible for each component of a blood cell to be used individually for various medical scenarios, however this study will not focus on the individual usage of blood components, but its focus would be on the overall WB unit. In accordance to the ABO blood system (a system used to classify the nature of an individual's blood type) [3, 46], blood within humans have four different blood groups, namely A, B, AB, O. With the introduction of a Rhesus (Rh) value which can either be positive (+) or negative (-), it ends up doubling the initial blood groups which results in eight different blood types. The different blood types play a vital role with regards to storage, and distribution. In terms of storage, certain blood types are considered as rare, as very few individuals carry this particular blood type, for example, the blood type AB^- is extremely scarce within the South African population as 1% of the total population carry AB^- blood. With regard to distribution, the relevance of blood types and compatibility plays an important role during transfusion.

Cases have arisen when medical knowledge was insufficient, where patients received incompatible blood types which have adverse health effects resulting in blood clumping (also referred to as agglutination), which can be life threatening.

The demand for WB units can be broken down into two main scenarios. The first scenario relates to premeditated medical events which are in need of WB units, this can involve events such as general surgeries. Premeditated events allow the blood bank to allocate adequate amounts of WB units towards the desired patient. In addition, if the blood banks have an inadequate amount of WB units to meet the demand, then it still leaves the blood bank with adequate time to import additional units from external sources. The second scenario relates to unforeseen patients who are in immediate need of WB units. Blood banks tend to struggle to fulfil this type of demand if large influxes of patients are in need of treatment involving blood units at the same time. Most blood banks tend to stock-pile WB units and other blood products during periods of expected sudden trauma, however this is not a solution as demand for WB units can still out-weigh the on-hand supply. The act of importing additional WB units raises the expenses incurred by the blood bank. Therefore, blood banks face the important challenge of controlling their blood product inventories [43]. Another factor which can contribute towards additional expenses relates to expiration of WB units. Expiry of units occurs when the WB units exceed the expected shelf-life, and the act of disposing the WB units in an appropriate manner increases the blood banks expenses. The act of freezing blood units can prolong the lifespan of the cells, however, this study will ignore such WB units. The proposed blood assignment method generally seeks to minimize wastage of blood products by efficiently assigning blood to patients and preserving blood stock pile by allocating available blood to different blood types.

The Blood Assignment Problem (BAP) is an optimization task which tries to efficiently assign WB units to patients in need, whilst trying to minimize the amount of importation and expiry within the blood bank. The BAP is comprised of several individual components which can demonstrate issues when trying to formulate a mathematical model for the problem at hand. Such components include daily demand, cross-matching of blood types, importing additional blood units and expiring WB units when these have exceeded their shelf-life. In addition, the research conducted in this study is from a perspective of the South African National Blood Service (SANBS) and in relation to the South African population.

1.2 Problem Statement

The demand received for WB units must be met. With this in mind, if the daily demand exceeds the current supply on hand, then additional WB units must be imported from external sources in order to satisfy the demand. Blood banks receive WB units which are largely contributed from individuals' voluntary donations. These fresh WB units enter a queuing system also referred to as the First-in First-out (FIFO) method; this implies that the oldest WB units would be used first with the newer units being placed at the back of the queue to be used at a later time. Figure 1 below illustrates the process of using the FIFO system for distributing WB units.

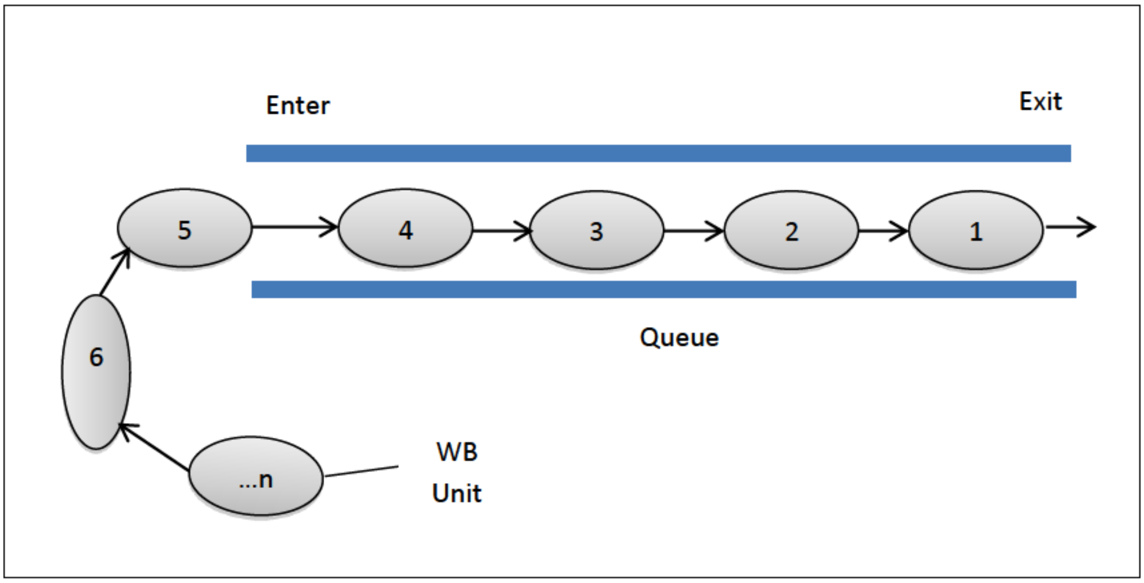

Figure 1: Illustration of the FIFO issuing process of WB units when exiting storage

In Figure 1, the oval shapes represent WB units, with the numbers ranging from 1 to n, where n represents any positive integer. The WB units proceed to follow the queue until it exits, thus implying that the WB unit is being distributed to a patient. Note that Figure 1 illustrates the queuing system for all blood types. The effect of implementing the FIFO system aims at minimizing expiry among WB units, as the oldest WB units are being utilised first, and thus decreasing its exposure to storage time. Any WB units that have exceeded their respective shelf-life are discarded from storage, which is a negative connotation for the blood bank as this incurs additional expenses and implies that the blood bank did not utilises resources in an effective manner. It would be appropriate for a patient to receive blood relating to their specific type, however a situation might arise where there is an inadequate supply of WB units pertaining to that specific blood type. When this occurs, various processes occur to try and satisfy the demand. Firstly, the request will try to be satisfied by using other compatible blood types. Each blood type must first try and satisfy the demand pertaining to their own types, once their respective demands have been met, only then can the act of compatibility distribution occur. Compatibility distribution in this study relates to using the remaining units from each blood type and redistributing those units to other compatible blood types. The act of redistribution minimizes additional blood importation from external sources, and uses resources more effectively. However, if the remaining WB units after redistribution cannot meet up to a day's demand, then additional WB units would be imported from external sources. The act of importing WB units incurs additional expenses for the blood bank, and must therefore be minimized. The BAP can be summarized into four major components:

i. Supply: Stock on hand at any given moment.
ii. Demand: Relates to both planned and unplanned requests for WB units.
iii. Importation: Utilizes additional WB units from external sources in order to satisfy a demand.
iv. Expiry: Occurs when WB units exceed their shelf-life.

Figure 2 is a flowchart depicting the steps taken each day within the blood bank for the management of WB units. In addition, it also illustrates the steps that must be fulfilled in order for additional WB unit importation to occur.

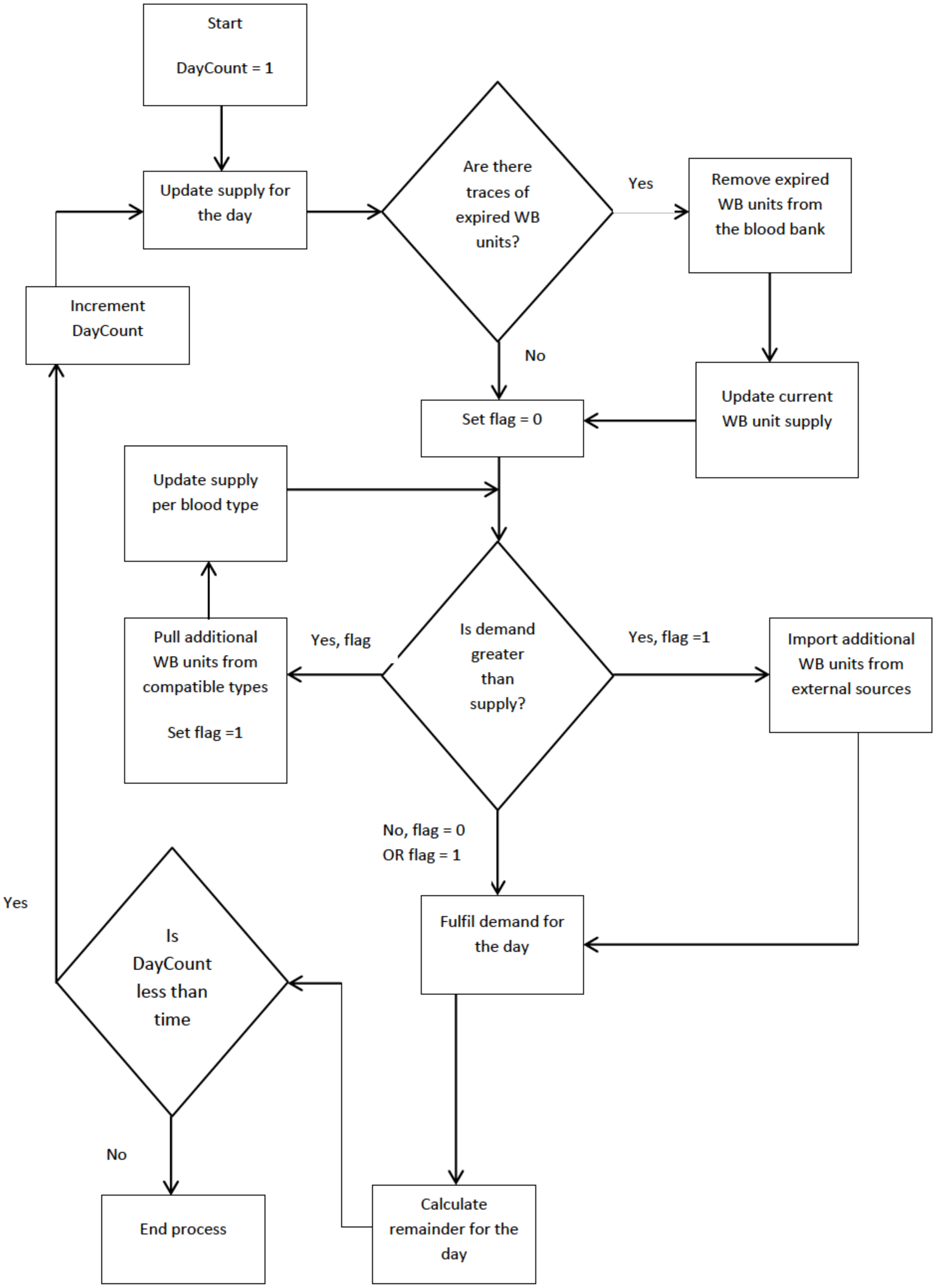

Figure 2: Flow diagram representing the daily processes that occur with the blood bank in the current study

1.3 Motivation

The notion of optimization can successfully be applied to various complex decision making or allocation problems [4]. The BAP is considered as a NP-hard problem, due to the complexities associated with blood assignment process such as the stochastic behaviour of daily demand, influx of supply and blood compatibility [5]. Prior research specifically dedicated for the BAP is relatively scarce, with only a handful of research tackling the problem from an optimization perspective. An optimized blood distribution system can be a difficult task due to the overwhelming factors that act upon the system [47], however blood is often the solution for patients who experience large amounts of blood loss during surgeries/trauma, anaemia or other blood related illnesses [48]. Therefore, the need for an efficient blood allocation system in blood banks is justified.

The following study of the BAP attempts to incorporate a previous mathematical model [7] and explores different metaheuristic algorithms to work the BAP. The algorithms implemented in this study include: Genetic Algorithm (GA), Particle Swarm Optimization (PSO), Duelling Algorithm (DA), Symbiotic Organism Search (SOS), and the Grey Wolf Optimizer (GWO). The BAP can also be seen as a perishable inventory problem, the current study can therefore contribute towards future research relating to blood and other relatable perishable inventory problems. Prior research suffered with the inability of using appropriate real-world datasets to examine their implementations for working the BAP. Due to confidentiality issues, this study also could not utilize real-world datasets. To confront this problem, datasets were stochastically generated, and in addition the current study attempted to utilise events and statistics prevalent to South Africa. By utilising such factors, the datasets (though randomly generated) can try to reduce such randomness when generating values for demand of WB units. The idea behind generating datasets using a statistical approach is based on social convention; the demand for WB units should have certain trends. For example, the demand for WB units should have a greater need during months that experience more public holidays or breaks from educational institutions due to higher levels of dangerous activity such as drunk driving and other criminal activities.

Expenses are an aspect that must be within any organization, and SANBS is no different. In accordance to [5], SANBS requires various equipment and chemicals in order to test and store WB units. Testing WB units is a vital component of any blood bank to ensure that the units are viable (do not contain harmful diseases), and do not transmit any pathogens to receiving patients. A study conducted by [6] emphasized the four main activities related to blood and blood product management which includes:

i. **Production costs:** Relate to costs of running blood campaign collections, also included in this amount will be procedures in related to testing the WB units.
ii. **Expiry costs:** WB units have a lifespan of between 30-35 days, once a WB unit has exceeded its lifespan, there are costs attached to incinerating the units.
iii. **Inventory costs:** Costs associated with keeping WB units at optimum temperatures and in an adequate environment.
iv. **Importation costs:** If in the event the supply at hand does not cover the demand for that day, additional WB units will have to be imported in order to fulfil the remaining demand.

The current study will focus only on costs related to expiry and importation, as by minimizing the objective function of the BAP, this will relate to lower costs incurred by the blood bank.

1.4 Aim and Objectives

The primary objective of this project is to obtain an in-depth understanding of the BAP inventory management process. The work done in this study attempts to offer content which could possibly advance the research of the BAP or similar perishable inventory related problems. In addition, this study implemented a more accurate method when generating stochastic datasets. The following are the specific objectives that will be accomplished in this study:

i. Development of a robust blood bank management policy which aims at efficiently using WB units.
ii. Implementation of five population-based metaheuristic algorithms to work the BAP. The algorithms include: GA, PSO, DA, SOS and GWO. These algorithms will be evaluated against each other and results will be documented in a meaningful manner which will aid in determining the superior implementation for the BAP.
iii. To generate a variety of stochastically based datasets which will examine how efficiently the blood bank is able to distribute WB units, and to test the levels of importation and expiry that might be faced. The datasets will embody various

situations that the blood bank may experience during the course of its existence. Some of the datasets will also provide a social approach in order reduce randomness with reference to demand generation. This social component will be based on a South African perspective, and will incorporate aspects like public holidays, and terms in educational institutions.

1.5. Scope and limitations

The BAP is comprised of various components such as WB unit deterioration, random nature of demand and supply as well as WB compatibility. Due to these components and other factors certain assumptions will be introduced in order to develop a mathematical model that would be adequate for the problem at hand:

i. Race, gender, age and other contributing traits that characterize an individual will be ignored. The focus of the study is solely based on blood units.
ii. All WB units will be deemed as “clean”, this means that no WB unit will contain pathogens.
iii. A year will consist of 365 days.
iv. The lifespan of a WB unit will be 30 days. This is in accordance to the study conducted by [7, 42].
v. The first day (day 1) will have no carryover of WB units from the previous day.
vi. All blood types will first fulfil requests associated with their blood types, from there, the remainder from each blood group can contribute to other compatible blood types.

An additional limitation relates to the use of datasets in this study. Due to confidentiality issues, datasets were randomly generated using percentage bounds to generate values for supply. The concept of percentage bound is elucidated in Chapter Three.

1.6. Methodology

The BAP is comprised of various components that were examined individually in order to formulate a mathematical model. The methodology followed a step-by-step pattern as various components relied upon the previous steps in order to proceed. Figure 3 illustrates a flow diagram of the steps conducted in this research of the BAP. The methodology consists of seven main steps which will be expanded upon in later chapters:

Step 1: Obtain an in-depth understanding of blood banking, and implement an appropriate policy for issuing WB units.

Step 2: Adapt the five metaheuristic algorithms namely, GA, PSO, DA, SOS and GWO to the chosen blood banking policy, in order to solve the BAP.

Step 3: Since the datasets are stochastically generated, certain restrictions must be followed in order to mimic real life scenarios that may be faced by the blood bank at any given moment.

Step 4: In order to test how effectively each Metaheuristic Algorithm works the BAP, each algorithm will be subjected to the datasets.

Step 5: Computational results were used to determine the superiority of each algorithm. The assessment of each algorithm was based on the results obtained in correlation to the objective function of the BAP (further discussed in Chapter Three).

Step 6: Record results in a manner that aids in identifying the better algorithm both analytically and visually.

Step 7: The records obtained from step 6 allowed for an extensive review of each metaheuristic algorithm which in turn revealed the best suited algorithm to work the BAP.

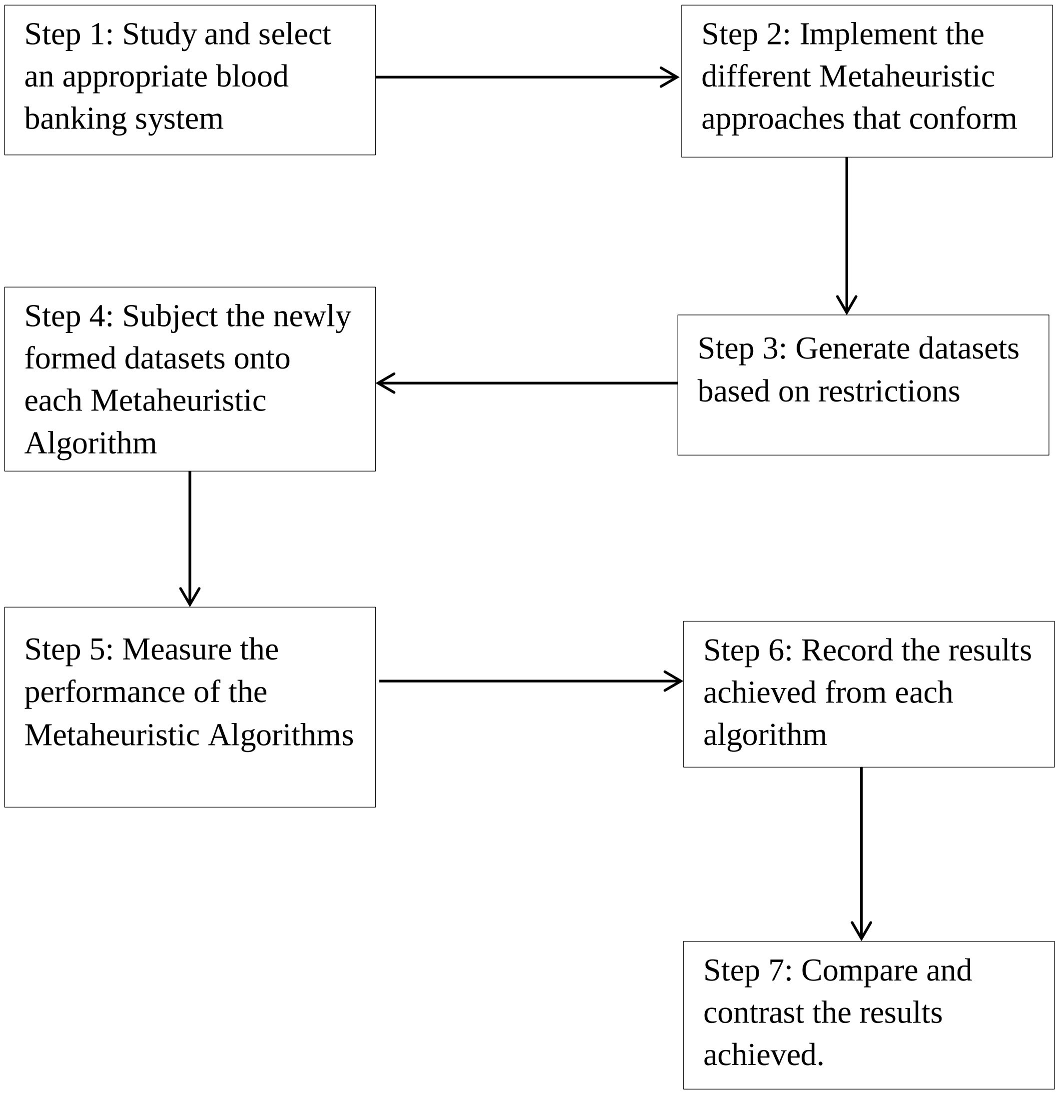

Figure 3: Flowchart representation of the steps conducted in this study of the BAP.

1.7 Thesis Layout

Chapter One presents the background of the study, motivation of work, problem statement, research objectives, and scope and limitation of the work. Chapter Two presents the review of related work on the BAP. Chapter Three describes the research methodology based on the BAP which includes the development of a mathematical model and implementation of the metaheuristics algorithm. Chapter Four discusses the results obtained for each metaheuristic in accordance to each dataset. Finally, Chapter Five includes the final remarks relating to the BAP in this study as well as future directions.

Chapter Two
Literature review

This chapter provides insight to previous literature that has studied the blood management process as well as research conducted in relation with the BAP. In order to understand the BAP, it is important to obtain an in-depth understanding with regards to individual components that compile the BAP. Due to the nature of data generation, it is also imperative to understand certain attributes which pertain to the South Africa population.

2.1 Blood Basics

Blood transfusions save thousands of lives on a daily basis by means of medical treatment [8]. Before divulging the mathematical and computational components of prior research, blood itself has been extensively researched with the aim of improving health care for individuals. Experiments relating to blood transfusions were conducted in several steps. First attempts of blood transfusion involved human’s drinking or bathing in donor blood [53], but as medical knowledge evolved transfusion attempts between animals were examined, and then from animal into man with the first evidence of a transfusion occurring around 1666 in Oxford [9]. Since then, the advancements of medical technology have assisted with the understanding of blood compatibility and transfusion. Using incompatible blood types for transfusion often resulted in blood clumping and other negative side effects upon the patient. In 1927, antigens A and B where discovered followed by antigen O which resulted in the widely known ABO blood grouping system [10]. With the introduction of Rh value (mentioned in Chapter One), this further resulted to eight different blood types found in humans. Table 1 shows an illustration of compatibility among blood types. Note that “YES” implies that the blood types are compatible, and “NO” indicates that they are not compatible.

Table 1: Representation of blood compatibility [7]

Blood Types	A+	A-	B+	B-	AB+	AB-	O+	O-
A+	YES	YES	NO	NO	NO	NO	YES	YES
A-	NO	YES	NO	NO	NO	NO	NO	YES
B+	NO	NO	YES	YES	NO	NO	YES	YES
B-	NO	NO	NO	YES	NO	NO	NO	YES
AB+	YES	YES	YES	YES	YES	YES	YES	YES
AB-	NO	NO	NO	YES	NO	YES	NO	YES
O+	NO	NO	NO	NO	NO	NO	NO	YES
O-	NO	NO	NO	NO	NO	NO	NO	YES

Even though blood is donated as whole units, the units can still be separated into its various components, these components can be used for treating medical conditions which pertain to the components' function (discussed in section 1.1). For example, blood platelets can be harvested and utilized for patients with bleeding disorders [11, 41]. The act of separating components for individual usage fully maximizes its utility at aiding patients with various health issues. Possible blood substitutes originated from the early 1600's [44], however [45] reported possible side effects with the use of artificial blood such as a negative response from the body's immune system.

WB units are stored in freezers at certain temperatures in order to prolong their lifespan. Depending on the anticoagulant (a solution used to preserve WB units) used within the blood unit, lifespan per WB unit may vary. Temperatures inside these freezers range between 2-4 degrees Celsius with a lifespan ranging between 28-42 days [12]. Cryo-preservation relates to freezing WB units in order to drastically prolong their lifespan and can be viable for up to 10 years [13]. However, the act of freezing can be time consuming and incurs additional costs by the blood bank. These frozen WB units also follow specific guidelines when thawing out and require sophisticated equipment during the Cryo-preservation process.

2.2 Overview of Metaheuristics

The following study implemented a diverse range of nature inspired metaheuristics in order to work the BAP. The list includes GA, PSO, DA, SOS and the GWO, each of which have been implemented in various optimization problems in the past. A Metaheuristic is defined as a "*high level problem independent algorithm framework that provides a set of strategies to develop heuristic optimization algorithms*", unlike normal heuristics which are problem dependent [14]. Though the algorithms vary with regard to structure and implementation, the overall goal of generating potential solutions and selecting the best outcome remains the same. A massive advantage of using a metaheuristic algorithm lies in its ability for the algorithm to find a good solution with minimal computational effort or iterative methods. This study opted to use metaheuristics from previous literature, such as the GA and PSO, as this study incorporates a different model for supply generation that differs from previous work. Therefore, it would be interesting to see the results obtained in this study and compare it to previous relatable work. Secondly, the remaining metaheuristics, DA, SOS, and GWO have not yet be examined for the BAP, and are therefore implemented in this study.

GA is an extensively researched algorithm with a wide-range of applications to solve complex optimization problems. According to the work presented in [15], the GA utilises sub-procedures in order to morph a potential solution so as to obtain an improved solution. The School Timetabling Problem (STP) has been broadly studied by various researchers with many implementations utilising the GA or a hybridized version to work the problem. The report conducted in [16] summarized various hybrid implementations and compared results obtained from the various hybrid GA algorithms for the STP. Another use of GA was exemplified in the Bus Driver Scheduling Problem (BSP). The study in [17] modified the GA by implementing a "pieces-of-work" coding scheme which was broken into two aspects, the first being the duties and secondly the pieces of work. The chromosome was created using each piece of work representing a gene. Dias's work showed a unique implementation of GA for solving the BSP, and proved to produce quick and satisfactory results.

The development of the PSO has been accredited to Kennedy [18]. PSO has been applied to many problems like Artificial Neural Network (ANN) training, Fuzzy Control (FC), Pattern Classification and Function Optimization. In recent years, PSO has received a lot of recognition due to its fast convergence rate and ease for implementing the algorithm to solve problems. The Job Shop Scheduling (JSS) problem is a well-known optimization problem, with the objective of minimizing completion time for all jobs to be completed, a study conducted in [19], applied PSO towards the JSS which resulted in a positive outcome of the PSO successfully achieving the JSS objective function.

The DA is an optimization algorithm which is based on how humans fight and learn from each other in order to improve their fighting capabilities [20]. The study conducted in [21] compared the GA and DA algorithms using two benchmarks which were maximization problems, results indicated that the DA approach achieved better and faster solutions, DA achieved solutions in 143 iterations, whilst GA needed 166 iterations. [21] further compared the DA algorithm to the PSO and Imperialist Competitive Algorithm (ICA), and discovered that DA achieved better results than PSO with regards to computational time.

The Parallel Machine Scheduling Problem (PMSP) is a combinatorial optimization problem which has been extensively researched. The problem consists of similar or unrelated types of multiple numbers of machines in which jobs can simultaneously be scheduled. In [22], an improved SOS algorithm was implemented and used to solve the PMSP with the objective of minimizing makespan. The SOS algorithm was originally introduced by Cheng in 2014 in which the algorithm was implemented to solve numerical engineering optimization problems [23]. The study conducted by [22] compared the results of the SOS implementation to previous work and found that the SOS algorithm outperforms for all the problem instances that were tested.

The Real Power Economic Dispatch (RPED) relates to allocating optimal power generation towards thermal units without violating constraints within the system. The RPED is considered as a non-linear problem [24]. In [24], the GWO algorithm was implemented and employed to solve the RPED, whilst examining the algorithms' effectiveness, robustness and feasibility. The tests were conducted with different kinds of constraints with results achieving minimum fuel expenditure.

Overall, metaheuristics have been implemented on a variety of complex problems. The literature mentioned above is just some of the studies in which GA, PSO, DA, SOS and GWO were examined. Even though metaheuristics differ with regards to implementation and structure, the overall aim relates to obtaining the best solution to solve a problem. The following research tries to contribute towards the study of these metaheuristics in reference to the aspect of medicinal optimization.

2.3 Blood management

Apart from the BAP, various components of blood management have been scrutinized over the years in order to make every aspect as efficient as possible. Blood management is defined as the appropriate use of blood and blood components with the aim of minimizing their usage [25]. The following section will have an in-depth analysis of prior research pertaining to WB and blood product management.

The study conducted in [4] focused towards minimizing the distance between blood centres and hospitals by means of graph partitioning coupled with metaheuristic algorithms, namely, Colliding Bodies Optimization (CBO). Results indicated that the proposed algorithm performed satisfactory in relation to optimal points of view and computational time. Another study which focused on blood bank locations was presented in [26], the study focused on three main costs which was in need of minimization, namely, fixed costs, delivery costs and emergency referral costs. The Capacitated Location Problem with Emergency Referral Model (CLPER) incorporates decision-making processes to determine the optimal number and locations for blood banks. In [26], the author formulated the CLPER as an integer programming model, and used real-world data from the Thai Red Cross Society. Results stated that the maximum distance between blood bank and hospital is 45 kilometres. The study in [27] incorporated a single and double allocation model which implied that the demand for blood can either be supplied from a single or from two blood banks with both models being represented by an integer programming model. Overall, the study indicated which blood banks offered the lowest delivery costs.

Studies pertaining to the BAP mainly analyse the overall blood bank issuing policy, and find ways to try and minimize expiration and importation of additional blood units from external sources. Research conducted in [3], [7], [8] and [28] followed the same overall structure within regards to mathematical model, but differed with regard to the metaheuristic algorithms implemented to work the BAP. Other similarities between the mentioned literatures lie with regard to the objective function of minimizing overall blood importation from external sources and dataset generation. In terms of datasets, data had to be randomly generated due to confidentiality issues when trying to obtain real-world datasets. The datasets examined varying situations which the blood bank could be exposed to. Such a situation includes a case where demand for blood units heavily exceeded supply, or vice versa.

The work presented in [3] conducted an in-depth study mainly pertaining to GA and hybrid GA implementations. The Metaheuristic Algorithms used within this study included GA, Adaptive Genetic Algorithm (AGA), Simulated, Annealing Genetic Algorithm (SAGA), Adaptive, Simulated Annealing, Genetic Algorithm (ASAGA) and the Hill Climbing (HC) Algorithm. In addition, the study also looked at the Simple Assignment Algorithm (SAA) in comparison to the Multiple Knapsack Assignment (MKA). The simple assignment approach states that blood units are set aside to meet the demand for the day, and once these units are set aside, they cannot be selected again. On the other hand, the MKA investigated if cross-matching between blood types can satisfy the demand in a day. Results indicated that all of the algorithm implementations achieved optimal fitness with the HC algorithm demonstrating greater results, whilst the MKA proved efficient in minimizing the amount of imported blood units. In [7], the PSO in conjunction with the MKA policy was implemented to solve the BAP. Also demonstrated in this research was the FIFO issuing system which was incorporated to minimize expiration. Results proved that the PSO implementation produced satisfactory results with low importation levels and no form of expiry. Another study conducted in [8] implemented different metaheuristic techniques, namely, the Tabu Search (TS) and Simulated Annealing (SA) to optimize the BAP, also included in this study was the hybridization between TS and SA. The hybrid implementation used TS in order to obtain the initial solutions to the BAP which was then passed onto SA for better exploitation. Results indicated that TS does not produce an efficient solution, whilst SA and the hybrid approach both fared well in optimizing the BAP. The final study that will be analysed relates to the work presented in [28], which involved the implementation of two local search techniques, namely, the Greedy randomized adaptive search procedure (GRASP) and Dynamic Programming (DP) in conjunction with the MKA. DP showed to import $O^{+}$and O^{-} blood rather heavily for the first 50 days until levelling out, whilst GRASP imported gradually as the days increased. GRASP did seem to cope much better than DP when demand exceeded supply for blood units. The results in [28] also concluded that population based algorithms tend to produce better results than local searches. A bi-objective integer programming approach was conducted by [60], who also analysed the stochastic nature of demand and supply, whilst incorporating importation to meet the demand for a day.

2.4 The South African Population

Due to this study incorporating certain statistics from the South African population for generating datasets, it is therefore necessary to study certain attributes that pertain to South Africa. South Africa is known for its diversity in culture and race, race referring to a person's ethnicity. South Africa has four common race groups, namely, Black, White, Indian and Coloured, with approximately 44 million people living in the country [29]. Figure 4 below is a pie chart representation of the race group percentages in South Africa.

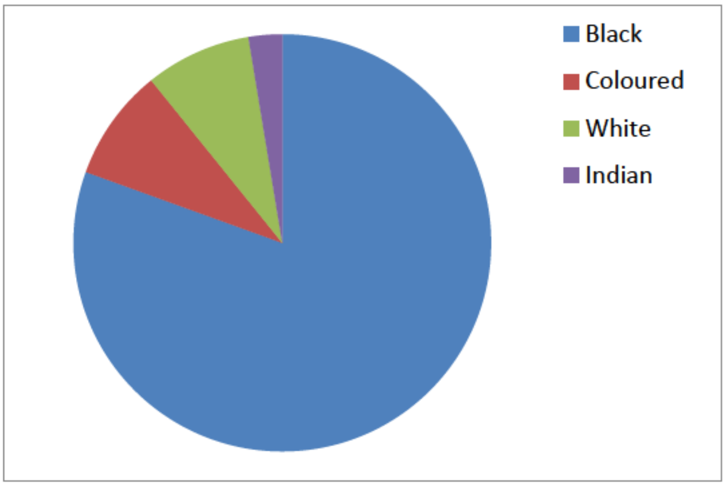

Figure 4: Pie chart representation of the current race group proportions in South Africa adapted from [7]

In accordance to Figure 4, the percentage of Black citizens in South Africa is 80.6%, Coloured is 8.7%, the White race group stands at 8.2% and finally the Indian race group is 2.6%. Race plays a role with regards to screening blood units before distributing it to future patients. Due to the Black race group being predominant within South Africa, it is therefore assumed that a larger proportion of blood diseases are linked to the Black community. In 2017, the percentage of Acquired Immune Deficiency Syndrome (AIDS) related deaths stood at 25.03% [29]. Human Immunodeficiency Virus (HIV) and AIDS are blood related diseases which break down an individual's immune system making them susceptible to other infections and illnesses [30]. Currently there is no cure for HIV/AIDS, with South Africa having the fourth highest HIV/AIDS prevalence rate of 18.9% [31].

Table 2: Illustration of the proportion of blood types found in the South African Population as adapted from [7]

Blood Type	A^+	A^-	B^+	B^-	AB^+	AB^-	O^+	O^-
Proportion (%)	32	5	12	2	3	1	39	6

Due to the variety of cultures in South Africa, the country also experiences a number of different public holidays. These holidays are derived from a variety of events, some of which are issued to honour the past of South African history, whilst others are culturally based. In addition to public holidays, educational facilities such as schools and tertiary institutes take mid-term breaks. The importance of these dates relates to the social behaviour aspect that will be represented in the BAP model. In theory, individuals indulge in more dangerous activities during months with more breaks and public holidays, therefore leading to an increase in demand of blood and blood products. Reports have shown that South Africa experiences an increase in the amount of drunk driving levels during Easter [32], therefore blood banks tend to stock-pile blood products as precautionary measures.

Table 3: Representation of the starting month and ending month of most educational institutions in South Africa [33].

Educational institutions terms	Start Month	End Month
1	January	March
2	April	June
3	July	September
4	October	December

Table 4: Representation of the public holidays in South Africa within a year

Date	Percentage bound (%)
1 January	New year's day
21 March	Human Rights day
14 April	Good Friday
17 April	Family day
27 April	Freedom day
1 May	Workers day
16 June	Youth day
9 August	Woman's day
24 September	Heritage day
16 December	Day of recognition
25 December	Christmas day
26 December	Boxing day

The current chapter has provided more in-depth knowledge pertaining to blood, metaheuristics, and blood management with relation to the South African population. It attempts to further the research for the BAP by implementing five population-based metaheuristics and comparing the results between the implementations as well as previous literature in order to establish the superior algorithm. The relevance of incorporating South African statistics tries to minimize unpredictability when generating stochastic datasets. Previous work dealing with the BAP also suffered with the issue of obtaining real-world datasets and therefore randomly generated their own data in accordance to certain restrictions. The method implemented in this study for generating data tries to bridge the gap from the previous work, and which can also contribute towards future work pertaining to inventory management problems.

Chapter Three

Research Methodology

Demand and supply for WB units follow a stochastic trend, even though this study tries to implement a method for randomly generating datasets based on a specific month. However, in the real world, mass demand for WB units may occur at any moment regardless of the month. On an ideal day the total value for WB unit demand will be equal to the supply on hand, this eliminates the possibility of expiration, and also prevents the blood bank from importing additional WB units from external sources. This chapter will explore the technical components that were implemented to work the BAP, which includes global optimisation implementations, the blood compatibility process, expiring old WB units, and importing additional WB units when demand exceeds supply for a day.

As mentioned previously, the demand and supply for WB units follows a stochastic trend. In an ideal day, the supply for each blood type would meet the exact demand level which in turn eliminates importation from additional units, as well as carrying over excess stock which opens the WB units to possible expiry. Therefore, several metaheuristic algorithms were implemented which randomly generates a demand and supply based on South African social trends with each implementation trying to find the best possible solution for the day. Inclusive of the algorithms, there are four aspects which are combined to offer a solution for the BAP, and consequently offer optimal WB unit assignment in relation to demand. The four components include the global optimisation implementations, the blood compatibility process, expiring old WB units, and importing additional WB units when demand exceeds supply for a day.

3.1 Mathematical model

3.1.1 Objective function

The objective function for the BAP is given by equation eq. 1. The aim relates to minimizing the total amount of importation of WB units, as well as to minimize the expiration experienced by the blood bank. The objective function will also be used as a measure of how well a metaheuristic algorithm managed to minimize the levels of importation and expiration eq. 1. Breaking down eq. 1 reveals two sub equations namely I_{total} and E_{total} eqs. 2 and 3, which represent the total levels of importation and expiry respectively on a daily basis. The combinations of both eq. 2 and eq. 3 result in the objective function of eq. 1. A low value for eq. 1 represents a positive result, as the blood bank would incur a lower expense relating to importation and disposing of WB units, likewise a higher value for eq. 1 is deemed as a negative result for the blood bank. A negative result implies two things about the blood bank, firstly WB units are not being utilised efficiently which could be linked to the blood banks issuing policy or an external factor relating to donors or patients, for example, donations exceeding requests may result in possible expiry, and requests exceeding donations will result in higher importation levels.

Let
I: Represent the amount of importation
E: Represent the amount of expiration
d: Represent the day

$$\text{Min: } \sum_{d=1}^{n} (I_{Total} + E_{Total})_d \tag{1}$$

Where:
$1 \leq d \leq 365$,

$$I_{total} = I_{A^+}(d) + I_{A^-}(d) + I_{B^+}(d) + I_{B^-}(d) + I_{AB^+}(d) + I_{AB^-}(d) + I_{O^+}(d) + I_{O^-}(d) \tag{2}$$

$$E_{total} = E_{A^+}(d) + E_{A^-}(d) + E_{B^+}(d) + E_{B^-}(d) + E_{AB^+}(d) + E_{AB^-}(d) + E_{O^+}(d) + E_{O^-}(d) \tag{3}$$

Demand constraint

$$D_d \leq V_d, \forall B \tag{4}$$

Supply constraints

$$R = (S_B)_d - (D_B)_d \tag{5}$$

$$S_d = R_{d-1} + S_d, \forall B \tag{6}$$

Importation constraints

$When\ (D_B)_d > (S_B)_d$

$$(I_B)_d = (D_B)_d - (S_B)_d \tag{7}$$

Expiration constraints

When $d > 30$

$$\text{If } \sum_{d=30}^{t} (D_B)_d < (S_B)_{d-30} \tag{8}$$

Then

$$(E_B)_d = (S_B)_{d-30} - \sum_{d=30}^{t} (D_B)_d \tag{9}$$

where $1 \leq d \leq 365$

Non-negativity constraints

$$I, E \geq 0 \tag{10}$$

Where:

t: Represents a finite time period

d: Day

B: Blood type

I: Represents importation across t

E: Represents expiration across t

S: Supply for any given day

D: Demand for any given day

R: Remaining blood units

V: Represents an initial blood volume available in the blood bank

Comments:

1. The Objective function is to minimize expiration and importation within a finite frame.
2. Relates to the total amount of importation per blood type.
3. Relates to the total amount of expiration per blood type.
4. Demand for WB units, must be less than or equal to the volume in the blood bank for all blood types.
5. WB supply minus WB demand equates to the remaining WB units per blood type, only when supply exceeds demand in a given day.
6. Add the remaining WB units from the previous day to the new influx of supply to obtain the supply value for the new day.
7. When demand exceeds supply for a certain blood type, import additional WB units from external sources.
8. Condition: If the demand summed over 30 days exceeds the supply value of (day-30).
9. Expiration equals to the demand summed over 30 days subtracted from the supply on (day - 30).

As mentioned previously, the BAP is constituted of various components which contribute towards its complexity (section 1.5). In order to work the BAP, various hard constraints were presented (eq. 4-10) which focused on such components.

- At the start of a new day, the supply level for WB units cannot equal 0.
- Demand must be satisfied on a daily basis. If the demand exceeds supply then the blood bank must utilise compatible units. If the demand still exceeds supply then extra units must be imported from external sources in order to meet the demand.
- A WB unit must not exceed its shelf-life of 30 days. If a WB unit bypasses this shelf-life it has expired, and must be removed from the blood bank.

- Pulling additional WB units from compatible blood types to meet the demand for a day must be done in descending order. In other words, remaining WB units from a blood type with a higher proportion (table 2) must be used first, followed by the second highest proportion, and so on.

3.1.2 Generating Demand and Supply

Due to confidentiality issues, it was not possible to use datasets from hospitals/clinics in this study. In order to test each implementation, values for both demand and supply had to be randomly generated. In order to generate more accurate values, this study incorporated South African social trends based on monthly statistics. Ideally the most adequate statistics would be related to monthly usage of blood products in the country, however these statistics could not be found. Instead this study will incorporate monthly holidays as well as school terms and breaks from other educational institutions. The idea behind this method tries to show that the demand for blood has trends associated with a specific month. For example, demand would be expected to have a higher value in a month like December due to many people being off from work, schools and other institutions, as well as the rise of dangerous events such as drinking and driving and criminal events. In South Africa, months like April or December experience more public holidays (refer to Table 3), resulting in more holiday-makers venturing on roads, and due to the influx of road-users, the rate of motor vehicle accidents increases, therefore blood banks tend to stock-pile blood products as a precautionary measure. Taking this and other social trends into account like educational institution terms (refer to Table 2), it is possible to allocate each month with a specified percentage range for generating a value for demand. There were no significant events apart from occasional blood drives for generating values for supply, therefore the supply bounds will be set between 25% and 75%, which are bounds adapted from previous research conducted in [3] [7] [28].

Table **5:** Illustration of the percentage bounded ranges used for generating demand.

Month	Upper and lower percentage bounds (%)
January	35-85
February	25-50
March	25-75
April	65-90
May	25-75
June	35-85
July	65-90
August	25-75
September	10-50
October	25-75
November	25-75
December	65-90

Due to no prior literature using such a technique for generating dataset values, the percentage ranges stated in Table 5 can be subjected to change depending upon the individual's perspective towards the situation. Ideally, the most appropriate form of statistics that can be used is the monthly request for WB units as recorded by the SANBS. However, this study tries to convey the concept of stochastically generating values, but to remove some of the randomness when generating amounts in order to obtain more accurate data. The percentages displayed in table 5 are formed from the statistics based of schooling terms and South African public holidays illustrated in tables 3 and 4 respectively.

Using the percentage bounds in Table 5, it is now possible to generate demand, as well as supply using eq. 12.

Let:
A: Represent the initial volume in a blood bank
d: Represent a day
m: Represent a month

b: Represent a blood type
Bu: Represent the upper percentage bound
Bl: Represent the lower percentage bound

$$Supply_b or\ Demand_b = A\,.\,(rng(B_U - B_l)_m) \quad (11)$$

From eq. 11, the supply or demand is generated by randomly selecting a percent between the upper and lower bounds depending on the month the system is currently in (this is established in accordance to the current day). This is then multiplied by the initial volume in a blood bank which generates a value for supply or demand. Once a value has been generated, the value is then split into 8 sub-values in accordance to Table 2, this accurately represents the quantity in accordance to blood proportion in the South African population.

3.1.3 Updating blood supply

The act of updating WB unit supply has two core components linked to it. Component 1 relates to daily donations received to the blood bank. Component 2 is the addition of the previous day's remainder added onto the new stock of the day. If the system is in the first day or there is no remainder from the previous day, then the remainder equates to 0. As suggested from Figure 1, the act of issuing WB units to patients follows the First-in-First-out (FIFO) principle, therefore any remaining units from the previous day will be issued out to patients first, whilst the newer units will remain in the queue until called upon. Eq. 12 states that the supply of blood type b equates to the sum of current supply plus the remainder value of b from day$(d-1)$.

Let
R: Represent the remainder
d: Represent a day
b: Represent a blood type

$$(Supply_b)_d = (Supply_b)_d + (R_b)_{d-1} \quad (12)$$

Where $d \geq 1$

3.1.4 Expiring Blood Units

WB units are considered a perishable commodity due to its limited lifespan. The WB units can be frozen to prolong its lifespan; however, this adds further costs incurred by the blood bank. This study neglects the use of frozen WB units, and sets expiration of these units to 30 days in accordance to the study conducted in [7]. This implies that a WB unit will be discarded if it is not used within 30 days of its first entry into the blood bank, as the blood cells would denature. The following algorithm states conditions that must be satisfied in order for expiry to occur. It is unlikely for expiry to occur when the daily demand and supply have similar levels or the daily demand exceeds the daily supply over a period of days, if these events occur then it is unlikely for a unit of blood to be on the shelf for 30 days. Below is Algorithm 1 which illustrates the pseudocode relating to the logic used when calculating expiration of WB units within the blood bank.

Algorithm 1: Pseudocode for Expiring WB units.

```
Setup control parameters: Supply:S, Demand: D,
Day: d, Expiration for the day: E, Demand summed over a specified period:Sumd,

        Begin Algorithm
        If d > 30 Then
                For i = 1 to 30 //30 represents the lifespan of a WB units
                        Sumd += D_d
                End For
                If Sumd < (S_b)_d − 30 Then
                 E = ((S_b)_d-30) – Sumd
                   //supply on (d − 30) – Sumd over 30 days equates to expiry
                End if
        End if

End Algorithm
```

Algorithm 1 is only executed after day 30, due to WB units having a lifespan of 30 days. Therefore, it’s impossible to have units expiring before this period, this also allows the implemented system to be more efficient.

3.1.5 Importing additional blood units

Importing additional blood units to meet the demand in a day poses further expenses by the blood bank. Logically minimizing the levels of importation would result in lower expenses, and would imply that the blood bank is utilising its resources efficiently. The amount of WB unit importation is dependent on the difference between the demand and supply within the day. For example, if the demand for a certain blood type is 10 units, and the current supply is only 5 units (after pulling from other compatible blood types) this would result in the blood bank importing an additional 5 WB units from an external source in order to meet the demand. Two circumstances have to occur before importation can take place, these include

I. Demand exceeding supply in a given day.
II. Demand still exceeds supply after additional blood units are pulled from compatible blood types.

If the two circumstances are satisfied, only then can additional blood units can be imported from external sources. In theory, importation should have higher levels in the first few starting days of a planning horizon, once an accumulation of certain blood types occur, importation starts to decline.

3.1.6 Bottom-up technique

When the WB units on hand cannot meet the demand for a day, additional units from other compatible blood types are used. Each blood type must fulfil their corresponding requests before distributing towards other compatible blood types. The bottom-up technique relates to a system which pulls from compatible blood types. Therefore, remaining units from a day are then split according to the number of possible compatible types. By implementing this technique, the blood bank will reduce importation of blood units, and utilise its resources more effectively, as compatible blood units will be distributed to patients rather than staying in storage. There are some medical cases which require the patient's specific blood type, however this study has chosen to ignore these occurrences. Unfortunately, this does hinder the deployment of a real-world system, but the aspect of blood specific cases could not be included due to inaccessible real world data.

Table 6**:** Representation of the blood types and the compatible blood types it can distribute towards (Caption seems to be incomplete)

Blood type	*Can distribute to*	*Splitting*
A^+	A^-, O^+, O^-	$\frac{(\text{Remainder}A^+)}{3}$
A^-	O^-	$\frac{(\text{Remainder}A^-)}{1}$
B^+	B^-, O^+, O^-	$\frac{(\text{Remainder}B^+)}{3}$
B^-	O^-	$\frac{(\text{Remainder}B^-)}{1}$
AB^+	A^+, A^-, B^+, B^-AB^-, O^+, O^-	$\frac{(\text{Remainder}AB^+)}{7}$
AB^-	A^-, B^-, O^-	$\frac{(\text{Remainder}AB^-)}{3}$
O^+	O^-	$\frac{(\text{Remainder}O^+)}{1}$
O^-	N/A	N/A

The process of utilising compatible blood is conducted in accordance to the proportion of blood types in South Africa (Table 2). The higher the proportion of a certain blood type results in that blood type distributing first, whilst a lower proportion results in the blood type being utilised last. For example, if A^+ requires additional units, it would first pull from O^+ blood which has a proportion of 39%, if the demand is still not satisfied, more blood units will be pulled from O^- and finally if the demand still exceeds supply more units will be pulled from A^-. After conducting the bottom-up technique, and if the demand still exceeds supply, then additional WB units will be imported. The act of pulling from compatible blood units in this manner tries to maximize the storage of blood types which have the lowest proportion (have a higher rarity). More common blood types also have a higher percentage of resupply.

3.1.7. Individual representation

Due to the limited number of different blood types in humans, it is therefore possible to create a finite individual of size eight (eight blood types) with each segment in the individual of a decimal type to take into account a relevant values for supply, demand, importation and expiration. The following figure represents a typical individual.

Position 1	Position 2	Position 3	Position 4	Position 5	Position 6	Position 7	Position 8
A^+	A^-	B^+	B^-	AB^+	AB^-	O^+	O^-

Figure 5: Representation of an individual used in the metaheuristic implementations

Each segment from Figure 5 stores a value which is calculated using the blood proportion percentage found in the South African population. By storing each value in a separate location, this allows for the manipulation of the individual in later calculations. For example, with reference to fig. 5 and table 2, if the initial blood volume within the blood bank is 300 units, then those 300 units will be split in accordance to the proportion of each blood type. In this example, 96 units will be allocated blood type A^+, and 15 units will be allocated to A^-. This process continues for the remaining of the blood types and finally concludes with allocating 18 units to type O^-.

3.2 Metaheuristic algorithms

This study implemented a variety of metaheuristic algorithms to work the BAP. In total five metaheuristic algorithms were implemented in conjunction with the mathematical components of Section 3.1. The metaheuristics include GA, PSO, DA, SOS and the GWO, prior research by [7] and [8] has already researched the effects of GA and PSO in correlation to the BAP. However, no previous research used the DA, SOS and GWO algorithms to solve the problem. Due to the nature of the BAP, some of the metaheuristic algorithms had to be tailored in order to accommodate the problem at hand, but still followed the overall structure pertaining to the algorithm. Using five algorithms also allowed for a more in-depth comparison between the results achieved per algorithm, thus allowing for a more comprehensive outcome with regards to the more superior algorithm for the BAP. The following section will explore each algorithm and different components which embody its implementation.

3.2.1 Genetic Algorithm

The GA implementation employs the use of populations consisting of individuals along with genetic operators such as selection, recombination and mutation [34, 59]. The genetic operators create diversity in a population in order to find the best possible solution to meet the demand for the day. This diversity also attempts to eliminate premature convergence. The population within this study utilized randomized individuals (Figure 5); in addition, a study conducted in [35], classified the GA process into five major components, namely, fitness function, population, selection, crossover and mutation. These five components were also incorporated in this study of the BAP. The structure of the individual (Figure 5) makes it possible to implement the genetic operators. The purpose of selection, crossover and mutation are explained below:

I. **Selection:** the process for selecting specific individuals from a population was conducted using tournament selection. A specified tournament size was established; from here the two individuals with the greatest fitness values are used for the next process.

II. **Crossover:** the two individuals that were previously selected are now subjected to the crossover process. Conventionally a crossover operator would select n (where $n > 0$) random crossover points in each individual and swap the genes accordingly. Due to the unique nature of the individual, swapping random points would result in inaccurate readings based on the blood percentages in the population. For example, a case could arise where the A^+ segment (which has a relatively high percentage) could swap with a lower percentage segment such as O^-. Due to this possibility, the current study implemented uniform crossover which selects n random points in both individuals and swaps their corresponding values. After conducting the crossover method, the algorithm is now left with two newly formed individuals, each of these individuals have their fitness calculated, with the fittest individual being chosen and subjected to the next step. Figure 6 depicts the crossing over method.

Figure 6 as presented below represents the mechanics for the crossing over algorithm used within GA, to promote diversity within a population. Note that Figure 5 illustrates the labels for each position which is applied to the individuals in Figure 6.

Random crossover positions: 1, 3, 7

Original Individual 1

48	8	18	3	5	2	59	11

Original Individual 2

57	6	22	2	6	2	47	8

After crossing over

Offspring 1

57	8	22	3	5	2	47	11

Offspring 2

48	6	18	2	6	2	59	8

Figure 6: Depiction of the crossing over effect between two individuals and the results obtained

III. **Mutation:** mutation alters part of the individual to obtain a newer individual. This study used point mutation, the process randomly selects n number of points in the individual, and recalculates the value at that position, the recalculation process will only occur if that value in the supply individual does not equal to the value in the demand individual. For example, if position 5 would be selected, the algorithm would then generate a new value for supply by initial random amount, and multiplying it by 39% (proportion of the blood type in South Africa).

Figure 7 as presented below depicts the effect of mutation on an individual. Note that Figure 5 illustrates the labels for each position which is applied to the individuals in Figure 7.

Mutation position: 3, 4

Original Individual 1

48	8	18	3	5	2	59	11

After mutation process

Mutated Individual 1

48	8	20	5	5	2	59	11

Figure 7: Illustration of the mutation process of an individual and the result obtained

Once these steps have been completed, the individual is then placed into a new population (regeneration process), with the cycle continuing until the maximum generation size is met, or a solution is found for the day. Below is algorithm 2 which illustrates the GA algorithm that was implemented in this study of the BAP.

Algorithm 2: Pseudocode for the Genetic Algorithm

Setup control parameters: Population size: n, Regeneration rate: R,
Mutation rate: M, Crossover rate: C
Begin
For i = 1 to n
Generate initial population
End For
While stopping criteria is false// Or maximum iterations is met
Implement selection based on point a
Implement crossover as depicted in Figure 6 in accordance with C
Implement mutation as depicted in Figure 7 in accordance with M
Evaluate individual using its fitness function
Generate new population
End While
Return best global individual
End

3.2.2. Particle Swarm Optimization

The Particle Swarm Optimization (PSO) is a population-based metaheuristic which utilises a swarm of particles to perform is optimization process [7, 56]. The PSO algorithm mimics behavioural patterns from animal groups that do not have a specific leader, such as a flock of birds [36]. The algorithm begins by randomly distributing particles in a solution space and then begins its iterative process to try and locate the best solution. The optimization process relies on communication between particles in order to establish movement of the particles within the search space. The particles utilise both the experience of itself, as well as reachable neighbouring particles to guide the searching process. Given an n-dimensional space, each particle is characterized by a position vector $X_i = (X_{i1} \ldots X_{in})$ as well as a velocity vector $V_i = (V_{i1}, \ldots, V_{in})$. Both X_i and V_i make use of the following equations to iteratively update themselves.

Let

Pi: Represents the best personal best position

Pg: Represents the global best position

$r1, r2$: Represents random values between [0, 1]

$c1, c2$: Represents scaling parameters.

ω: Represents the inertia weight

t: Represents the iteration index

$$V_i(t+1) = \omega V_i(t) + c_1 r_1 (P_i - X_i) + c_2 r_2 \left(P_g - X_i\right) \tag{13}$$

$$X_i(t+1) = X(t) + V_i(t+1) \tag{14}$$

A further look at the parameters, in correlation with eqs. 13 and 14 reveal that $c1$ and $c2$ exert random forces in the direction of both Pi and Pg, whilst the ω value aids in regulating the velocity which in turn helps to balance global and local searches. In this paper the PSO system was customized in order to conform to the BAP. For example, considering the PSO particles and particle positions; the study conducted by [7] used a string representation (using letters corresponding to blood types) for the daily demand and supply, and transformed the string into values to represent certain components in equations (13) and (14). This current study opts out of a string representation, instead using the individual representation in Figure 3, it is possible to sum each segment to obtain a value which can relate to either Xi, Pi and Pg.

Algorithm 3: Pseudocode for the Particle Swarm Optimization

```
Setup control parameters: Swarm size: s, Problem Dimension: d,
Search range: X_min, , X_max, Velocity range: V_min,, V_max, Inertia weight: ω

Begin Algorithm
    Initialize position and velocity for all particles in problem space
    While stopping criteria is false do
        Compute ω
        For i =1 to s
            For j=1 to d
                Update velocity using eq. 6 and check boundary using
                Update position using eq. 7 and check if boundary is valid.
                Compute f(X_i)
        End for
            If f(X_i) < f(pbest) Then
            pbest ← X_i

            If f(X_i) < f(gbest) Then
            gbest ← X_i

        End for
    End while
Return gbest
End Algorithm
```

3.2.3. Duellist Algorithm

The Duellist Algorithm (DA) is based on the GA approach, which was inspired by human fighting and the ability of learning [20, 55]. With the DA approach, all the individuals within a population are referred to as duellist, with the aspect of fighting to determine champions, winners and losers within the population. Unlike the GA approach which produce blind solutions (blind solutions relate to individuals being produced that may not be a better solution), the DA subjects' loser individuals to learn from the winner which tries to minimize the blind effect. A winner between two individuals is based on the physical nature of an individual (fitness value) as well as a luck coefficient (LC), which is a randomly generated value. The DA implements several steps before conducting a duel between two individuals:

I. **Pre-Qualification:** if the duellist is above a set fitness level, then the duellist is not selected to duel. This reduces computational time as duellists that are deemed as unfit will automatically be rejected due to the fact that they will not be able to attain a positive outcome when duelling with another individual.

II. **Board of champions:** the board of champions aims at keeping the best duellist in the competition. The purpose of the champion is to train newer duellists to compete against each other. If the newer duellist has a better fitness than the original champion, then the duellist swaps positions with the champion.

III. **Duelling schedule:** the schedule between two duellists is set randomly, with each duellist using their fighting potential as well as LC to determine a winner. Conventionally, the higher the fighting potential and LC coefficient results in an individual having a greater chance of becoming a winner. In accordance to the BAP, the best solution is considered to be the individual with the lowest fitness value, therefore to adapt the DA in conjunction with the BAP, the inverse function of the randomly generated LC value is added to the fitness using the following equations and algorithm:

Algorithm 4: Pseudocode for the duellist outcome after the fighting process

Setup control parameters:$DuellistA$, $DuellistB$, Luck coefficient: LC

$A\ (Luck) \leftarrow [A(Fighting\ Capabilities)\ \cdot\ (LC\ +\ (rand(0-1)\ \cdot\ LC))]^{-1}$
$B(Luck) \leftarrow [B(Fighting\ Capabilities)\ \cdot\ (LC\ +\ (rand(0-1)\ \cdot\ LC))]^{-1}$

If $(A(Fighting\ Capabilities)\ +\ A(Luck) \leq B(Fighting\ Capabilities)\ +\ B(Luck))$ **Then**

$Winner \leftarrow DuelistA$
$Loser \leftarrow DuelistB$

Else
$Winner \leftarrow DuelistB$
$Loser \leftarrow DuelistA$
End If

Algorithm 4 illustrates the process of establishing a winner and loser when conducting a duel between two individuals. Each duellist utilizes a luck coefficient which aids them during a duel.

IV. **Duellist improvement:** after conducting the duel, the duellists are categorized either as a winner, loser or champion. The loser and winner are then treated to a form of learning in order to improve representations. The loser learns from the winner, whilst the winner trains himself by randomly regenerating values for each segment only if that segment does not match the demand for a day, in hopes that the new result is better than the previous value. Since the demand and supply for a day follow the same individual representation, if the segment in the winner or loser individual matches the demand segment, then the segment does not change.

Algorithm 5: Pseudocode for the Duellist Algorithm

```
Setup control parameters: Amount of duellists: n, Champion: C
Begin
    For i = 0 to n
        Register duellists in tournament
    End For
    Determine C
    While stopping criteria is false do //Or maximum iteration is met
        Duel between duellist A and B
        Determine winner and loser
        Winner trains further
        Loser learns from winner
        Eliminate duellist with worst fitness
            If (fitness (Winner) < fitness (C))
                C = Winner
            End If
    End While
Return C
End Algorithm
```

3.2.4. Symbiotic Organism Search

The Symbiotic Organism Search (SOS) algorithm simulates the interactive behaviour of organisms within nature [37]. A notable advantage of the SOS algorithm is that it does not require any specific parameter tuning [38]. With reference to task scheduling optimization problems, the SOS provided near optimal solutions, and has since drawn attention to its capabilities for other optimization tasks [57]. A standard SOS algorithm is broken into three main categories, namely, Mutualism, Commensalism and Parasitism, each of these phases alters an individual(s) within a population attempting to obtain a better solution than its original representation. Unlike the GA algorithm which procreates individuals, the SOS tries to adapt individuals through a series of phases:

I. **Mutualism**: Organisms interact with each other in a way that benefits both parties. In other words, the individuals do not experience any form of hazardous behaviour which could threaten an individual's integrity. Eqs. 15, 16 and 17 convey the act of mutualism between two individuals.

Let Xi and Xj represent two random individuals within a population, and MV represent the Mutual Vector.

$$Xi_{\text{new}} = Xi + rand(0, 1) \cdot (X_{\text{best}} - MV \cdot BF1) \tag{15}$$

$$X_{\text{jnew}} = X_{\text{j}} + rand(0, 1) \cdot (X_{\text{best}} - MV \cdot BF2) \tag{16}$$

Where:

$$MV = (X_i + X_j) / 2 \tag{17}$$

The value obtained from $(X_{\text{best}} - MV)$ tries to increase survival in the population, with all improved individuals replacing the original individuals.

II. **Commensalism**: Organisms interact with each other in a way that results in one organism benefiting without harming or altering the other organism. Selection of two organisms is done randomly from the population, and have their fitness values evaluated, the fitter individual is labelled as X_i and the inferior individual is labelled as X_j.

$$X_{inew} = X_i + rand\ (-1, 1).\ (X_{best} - X_j) \tag{18}$$

X_i benefits from X_j by means of $(X_{best} - X_j)$ (19)

III. **Parasitism**: Organisms interact with each other in a way that benefits one organism (parasite) whilst harming the other organism (host). To evaluate a form of parasitism for the BAP, two individuals from a population are randomly selected, with each of its fitness values evaluated as being similar to the commensalism phase. Following the evaluation, the fitter individual is labelled as the parasite, and the inferior as the host.

The parasite then swaps segments of its representation with the host only if the value (from the host) improves its original solution. Algorithm 6 illustrates the SOS procedure, the algorithm was tailored to incorporate the BAP with the parasite section returning the best solution.

Algorithm 6: Symbiotic organism search algorithm

Setup control parameter: Ecosystem size: n, Host: H, Parasite: P

Begin Algorithm
Generate initial population of blood types X={$X_1, X_2, \dots, X_n$}, and evaluate its fitness
  While stopping criteria is false //Or maximum iterations is met
    For $i = 1 : n$
    Calculate fitness of each individual organism (blood types)
    X_{best} = individual with lowest fitness
    End For
    //Implement the three SOS interaction phases
    Mutualism phase (section 3.2.4 I)
    Commensalism phase (section 3.2.4 II)
    Parasitism phase (section 3.2.4 III)
    //**Begin Parasitism Phase**
    for i = 1 to length (P)
      if (P[i] is not equal to $demand$ [i]) **Then**
        $store\ diff1 = demand\ [i] - H\ [i]$
        $store\ diff2 = demand\ [i] - P\ [i]$
      end if
      if ($diff1 < diff2$) **Then**
          swap host and parasite segments.
      end if
      if ($diff1 \geq diff2$) **Then**
          do not replace value.
      end if
    // End Parasitism Phase
      If (fitness (Par) < fitness (X_{best})) **Then**
        $X_{best} = Par$
      End If
  End While
Return X_{best}
: End Algorithm

3.2.5. Grey Wolf Optimizer

The GWO was inspired from the canine family, with wolves being considered as apex predators (top of the food chain) [39, 58]. The algorithm is moulded around a pack of wolves (the pack ranging between 5- 12 wolves), with each pack containing 3 key members with the first, second and third also referred to as the alpha (α), beta (β) and delta (δ) respectively. Each of these wolves have their own tasks within the pack, for example the α is tasked with leading the pack, β wolf aids the α in decision making and is second in command, with δ wolves having to submit to α and β wolves. The lowest ranking wolves are referred to as omega wolves, which are considered the scapegoats of the pack. The GWO is defined as a predatory space of artificial wolves contained in a $N \times D$ where N is the number of wolves and D is the amount of variables of the BAP. The i^{th} position of a wolf is represented by $Xi = (X_{i1}, X_{i2}, \ldots, X_{iD})$ with X_{iD} is the dth variable value of the ith artificial wolf. Each value for X is represented by the sum of the supply for the day, with the summed demand value representing the prey. According to the study in [40, 54], the hunting patterns of grey wolves follow a certain pattern:

- Tracking, chasing and moving towards the prey,
- Encircling, and harassing the prey until it stops moving, and,
- Moving forward to attack the prey.

Using this information, the GWO can be broken into individual components and mathematically modelled.

I. **Wolf pack hierarchy:** the fittest individual within the pack will be deemed as the α wolf, likewise the second and third fittest individual will be the β and δ wolf respectively, whilst the remaining wolves will be deemed as the ω. The GWO algorithm utilises this hierarchy in order to conduct the optimization process.

II. **Encircling the prey:** as mentioned previously, wolves encircle their prey during a hunt. The following equations are proposed for calculating the encircling behaviour.

Let A and C represent coefficient vectors.

$$\vec{C} = 2 \, . \, \overrightarrow{r2} \quad (20)$$
$$\vec{A} = 2\vec{a} \, . \, \overrightarrow{r1} \, . \, \vec{a} \quad (21)$$

Where $\vec{a}$ is linearly decreased over a set number of iterations from 2 to 0. Whilst $\overrightarrow{r1}$, $\overrightarrow{r\ 2}$ represent random vectors between 0 and 1

$$\vec{D} = |\vec{C} \, . \, \vec{X}_{p}\,(t) - \vec{X}\,(t)\,| \quad (21)$$

$$\vec{X}(t+1) = \vec{X}_p(t) - \vec{A} \cdot \vec{D} \tag{22}$$

Where t indicates the iteration, $\vec{X}_p$ is the position vector of the prey and $\vec{X}$ represents the position vector of the hunter (grey wolf).

III. **Hunting:** the hunting component utilises the alpha wolf to lead the hunt, the beta and delta wolf may part-take in the hunt occasionally. It is assumed that the alpha, beta and delta wolves have better knowledge regarding the prey than the omega wolves. Due to the alpha taking the lead in the hunt, we assign the best candidate solution to the alpha wolf, and in ascending order of fitness, allocate the remaining candidate solutions to the beta, delta and omega wolves. After allocation of candidate solutions, the wolves then iteratively update their positions using the following equations.

$$\vec{D}_\alpha = |\vec{C}_1 \cdot \vec{X}_\alpha - \vec{X}| \tag{23}$$
$$\vec{D}_\beta = |\vec{C}_2 \cdot \vec{X}_\beta - \vec{X}| \tag{24}$$
$$\vec{D}_\delta = |\vec{C}_3 \cdot \vec{X}_\delta - \vec{X}| \tag{25}$$

$$\vec{X}_1 = \vec{X}_\alpha - \vec{A}_1 (\vec{D}_\alpha) \tag{26}$$
$$\vec{X}_2 = \vec{X}_\beta - \vec{A}_1 (\vec{D}_\beta) \tag{27}$$
$$\vec{X}_3 = \vec{X}_\delta - \vec{A}_1 (\vec{D}_\delta) \tag{28}$$

$$\vec{X}(t+1) = (\vec{X}_1 + \vec{X}_2 + \vec{X}_3) / 3 \tag{29}$$

IV. **Exploitation (Attacking process):** by decreasing the value of $\vec{a}$ over a set number of iterations, this mimics the process of a wolf approaching the prey. According to eq. (21), $\vec{a}$ is a component in calculating $\vec{A}$ which in turn decreases $\vec{A}$. The values for $\vec{A}$ lies between [-1, 1] with each position of the search agent lying between this specified range, if $\vec{A}$ is less than 1, this can be deemed as the wolf moving towards the prey.

V. **Exploration (Searching for prey):** as wolves search for prey, they tend to diverge from the pack, and then converge during an attack. The divergence pattern can be calculated using the value of $\vec{A}$ bounded between [-1, 1] this allows for global exploration to take place. Exploration is also favoured by component $\vec{C}$ which contains random values between [0, 2]. Component $\vec{C}$ allocates random weights to the prey in order to emphasize ($C > 1$) and de-emphasize ($C < 1$). As mentioned previously, $\vec{A}$ is linearly decreased by $\vec{a}$, and C is assigned random values to emphasize the exploration process. Component C can also be interpreted as naturally occurring obstacles which occur during a hunt. Depending on the position of the wolf, C can randomly give the prey a weight which either makes it easier or harder for the wolf to catch the prey.

VI. **General implementation:** the GWO starts with an initial random population of grey wolves (candidate solutions) which in correlation with the BAP, are represented as the supply of blood units for the day. The demand for WB units in a day is interpreted as the prey. The position of each wolf and prey equates to the sum of the values of each segment within the individual's representation (Figure 3). Using the equations stated in (20-29), each candidate solution iteratively updates their position from the prey, with the parameter $\vec{a}$ being linearly decreased from 2 to 0 over a number of iterations. Candidate solutions tend to converge towards the prey when $|\vec{A}| < 1$ and diverge when $|\vec{A}| > 1$. The termination criteria terminate when the max number of iterations have been reached, or the supply equates to the demand for a day. The reduced amount of search parameters of the GWO implementation is an important advantage of the algorithm [29]. Algorithm 7 is a pseudocode which illustrates the GWO implementation used in this study of the BAP.

Algorithm 7: Pseudocode for the Grey Wolf Optimizer

Setup control parameter: Pack size: n, Coefficient vectors: $\vec{A}$ and $\vec{C}$, Best individual: α,
Second Best individual: β, Third Best individual: δ

Begin Algorithm
    For length of pack size **Then**
        Generate initial population
        Evaluate fitness of each individual
    End for

    Assign α
    Assign β
    Assign δ

    While stopping criteria is false//Or maximum iterations is met
        $\boldsymbol{While}\ i = 0$ to n **do**
            Update search agents using eq. 23
            Decrement parameter $\vec{a}$
            Update coefficients $\vec{A}$ and $\vec{C}$using eq. 13 and 14
            Evaluate fitness of each search agent
            Increment i
        End While
        Update α, β, δ
    End While

Return α individual

End Algorithm

3.3 Data generation

Information obtained from the rate of demand over a prolonged period of time can be useful for creating a data bank which will aid decision making, in addition it offers a way for maximizing efficiency and service delivery [49]. Some of the previous work relating to blood management have managed to obtain real-world datasets [50, 51, 52] in the form of case studies that occurred within a hospital. In order to evaluate the effectiveness of a system, it would be ideal to obtain real-world data, however stochastic datasets have been used as substitutes in cases where real-world data proved difficult obtainable.

As mentioned previously, datasets in this study were generated stochastically as real-world datasets could not be attained. As such, previous work done by [7, 8, 28] constructed datasets using ranged percentage bounds and subsequently randomly selected a percentage between the bounds. This study also adopted the same principles and used a Random Number Generator (RNG) in order to generate the datasets. The datasets were run over a course of 1 year (365 days) and generated values for supply and demand of WB units. The blood bank can also experience certain situations which it must deal with for example, the demand for WB units could exceed the supply or vice versa. The following table lists the different datasets used for testing each metaheuristic algorithm. Note that SAGV represents South African Generated Values.

Table **7**: Representation of the datasets 1-6 used in this study of the BAP

Dataset	Initial Blood Volume	Demand bounds (%)	Supply bounds (%)
1	500	25-75	25-75
2	500	SAGV	25-75
3	500	75-100	25-50
4	500	25-50	75-100
5	5000	25-75	25-75
6	5000	SAGV	25-75

In accordance to Table 7 the following offers a more detailed explanation relating to each dataset, and the purpose for using those specific percentage bounds.

I. Dataset 1

Dataset 1 was regarded as the control dataset, and was used as a comparison to other datasets. Previous literature pertaining to the BAP used similar percentage ranges and initial blood volumes, therefore it seemed appropriate to evaluate the metaheuristic algorithms with these parameters.

II. Dataset 2

Apart from the different metaheuristic algorithms that were implemented to work the BAP, this study also incorporated a unique method for generating demand based on South African statistics (Section 2.4). Dataset 2 represented this ideology and tries to minimize the stochastic nature when randomly generating values for WB unit demand. The SAGV illustrated in Table 5 correlate to percentage bounds mentioned in Table 7.

III. Dataset 3

The blood bank can face a situation where the demand for WB units heavily exceeds the current supply on hand. Dataset 3 tests this scenario, and it was expected for the importation levels to increase radically in order to satisfy the demand. More importation results in a greater expense being incurred by the blood bank, therefore dataset 3 tested how well an algorithm coped with a higher demand volume.

IV. Dataset 4

Dataset 4 tests the opposite of Dataset 3. In this scenario the blood bank faces an excess supply of WB units with minimal demand. It would be expected for the expiry levels to increase as the more stock of WB units will remain on the shelf and not be used within its lifespan. This also incurs additional expenses by the blood bank and implies that the blood bank is not utilising its WB stock efficiently.

V. Datasets 5 and 6

Datasets 5 and 6 were a replica of Datasets 1 and 2 respectively. However, they differ by means of initial WB volume being 5000 units instead of 500 units. This dataset examines how well the metaheuristic algorithm copes with larger volumes of WB units.

3.4 Parameter Setting

The following section illustrates the parameters used per Metaheuristic implementations. In order to keep this study valid, population size, and maximum iterations were set to 50 and 1000 respectively per Metaheuristic Algorithm. Due to the varying structure of each algorithm, individual parameters were required. Previous work [7], [8] utilized specific values for their parameters pertaining to the GA and PSO. Since the GWO, SOS and DA implementations were not previously subjugated to the BAP, parameter setting followed previous work ascertaining to other forms of research. Note that RNG represents a Random Number Generator.

I. GA
- Generation rate: 25%
- Crossover rate: 70%
- Mutation rate: 5%

II. PSO
- Swarm size: 50
- $c1$ and $c2$: 1.7

- ω: 0.715
- $r1$ and $r2$: RNG [0, 1]

III. DA

- Luck Coefficient = 0.01

IV. SOS

- No form of parameters

V. GWO

- α : 2
- $\overrightarrow{r1}$ and $\overrightarrow{r2}$: RNG [0, 1]

This chapter aimed at providing an insight to the metaheuristic algorithms used in this study, and reported the stochastic datasets used to examine each algorithm. As mentioned earlier, the parameter setting for each metaheuristic algorithm was adapted from previous literatures, and to keep each result valid, each metaheuristic implementation was examined over a set population size of 50 with a maximum iteration of 1000.

Chapter Four

Experimental Results

The previous chapter gave an overview of the mathematical model and metaheuristic algorithms used in this study of the BAP. The following chapter will offer line graphs and averages attained in association with each metaheuristic. The objective function relates to minimizing the overall importation and expiration of blood units, whilst ensuring that all blood demands are met within a day. In an ideal day the request levels would be identical to the stock on hand which is deemed as a solution. A solution being located implies that no form of importation can occur, and the chances of possible expiry are reduced. The chances of finding a solution was higher at the start of the year as stock-piling might have taken some time to come into effect. Once a system was in the stock-piling phase, then the chances of obtaining a solution largely decreased. This chapter also provides an in-depth comparison between each algorithm in order to establish which algorithm performed the best for the particular dataset. Note that “MT” refers to metaheuristic.

4.1 Experimental setup

Each of the Metaheuristic Algorithms was implemented on Intel core i5 CPU with 2.5GHz and 4GB RAM and Windows 10.0 Operating system, while the implementation software was Java. Each of the algorithms was subjected to the datasets mentioned in Table 7 in order to obtain an output. Four variables were recorded, namely, demand, supply, importation and expiration, an algorithm would be deemed as effective if it amasses low amounts of expiry and importation of WB units. In total, five metaheuristics were subjected to six datasets resulting in 30 outcomes, all of which were tabulated and graphed accordingly. Since the remainder from the previous day was being added to the donations received by the current day, it was unlikely for a solution to be found. With this in mind the results were evaluated using three different aspects

- Running time
- Average amounts of both importation and expiration
- Time taken before stock-piling occurred

Running time relates to the duration for an algorithm to run over a 365-day period, the smaller the running time the better the algorithm. The average amounts for both expiration and importation correlates to the objection function mentioned in Section 3.1.1. Finally, the time taken for stock-piling to occur drastically reduces the levels of importation, and can therefore be seen as a positive effect. Stock-piling does increase the risk of possible expiry, but due to the lifespan of a WB unit being 30 days, and the manner in which data is generated, it is unlikely for a blood unit to exceed this time frame.

4.2 Results and Discussion

4.2.1 Dataset 1

Dataset 1 posed as the control dataset as previous literature used similar percentage bounds when generating demand and supply values. A control dataset establishes a baseline of results that can be compared against other datasets in order to attain similarities and differences. Dataset 1 used percentage bounds ranging between 25% – 75% for generating both demand and supply.

Table 8: Average results achieved for each metaheuristic implementation subjected to dataset 1 for each blood group measured in units.

MT	Variable	A^+	A^-	B^+	B^-	AB^+	AB^-	O^+	O^-
GA	**Supply**	40.00	6.25	15.00	2.50	3.75	1.25	48.75	8.75
	Demand	192.81	88.67	78.67	35.41	15.27	6.36	131.83	87.80
	Import	0.00	0.00	0.08	0.01	0.28	0.01	0.02	0.00
	Expiry	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
PSO	**Supply**	162.7	7.25	61.35	3.26	16.16	12.90	47.51	9.94
	Demand	40.51	6.33	15.19	2.53	3.80	1.27	49.37	8.86
	Import	2.81	1.71	0.67	0.70	0.48	0.02	14.90	2.39
	Expiry	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
DA	**Supply**	295.00	133.40	120.90	56.00	26.50	11.00	176.20	136.40
	Demand	39.27	6.14	14.72	2.45	3.68	1.23	47.86	8.59
	Import	0.45	0.03	0.17	0.00	0.25	0.02	0.35	0.02
	Expiry	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
SOS	**Supply**	192.81	88.67	78.67	35.41	15.27	6.36	131.83	87.80
	Demand	40.00	6.25	15.00	2.50	3.75	1.25	48.75	8.75
	Import	0.00	0.00	0.08	0.01	0.28	0.01	0.02	0.00
	Expiry	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
GWO	**Supply**	60.07	30.10	20.73	6.41	1.76	1.51	68.73	16.79
	Demand	40.32	6.30	15.12	2.52	3.78	1.26	49.14	8.82
	Import	1.36	0.01	1.07	0.09	2.02	0.26	1.14	0.00
	Expiry	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00

Table 9: Representation of the running time per metaheuristic for dataset 1

Metaheuristic	Time (Ms)	Time(Minutes)
GA	4748287	79.14
PSO	504291	8.40
DA	4899566	81.66
SOS	4776186	79.60
GWO	4752407	79.21

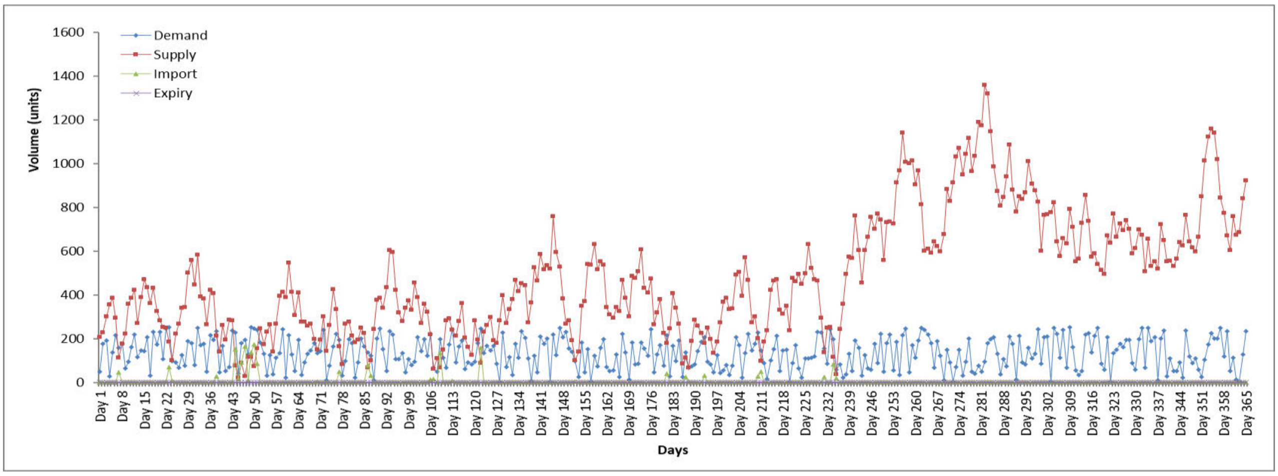

Figure 8: Representation of a line graph over a period of 365 days for the GA implementation of dataset 1

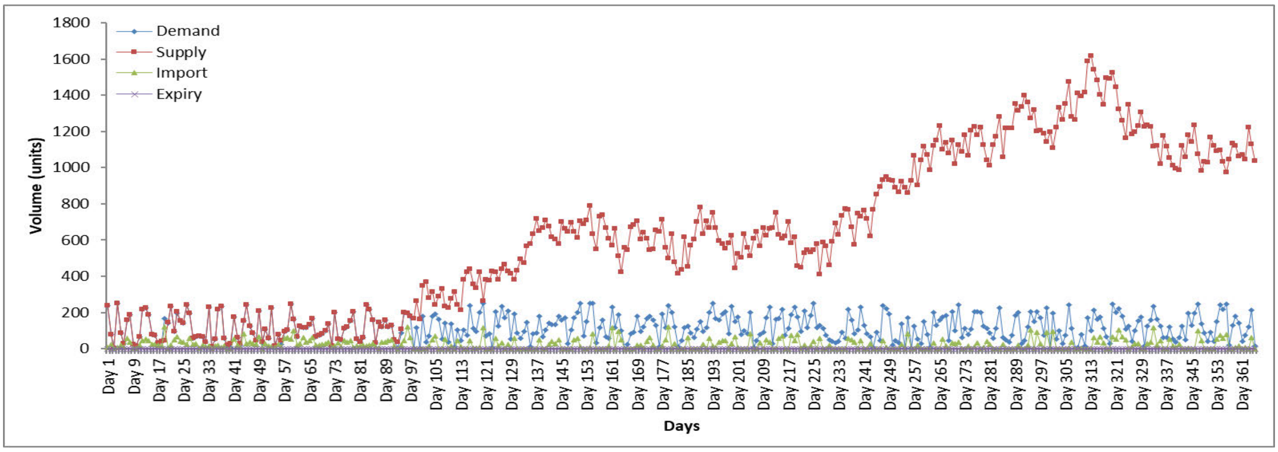

Figure 9: Representation of a line graph over a period of 365 days for the PSO implementation of dataset 1

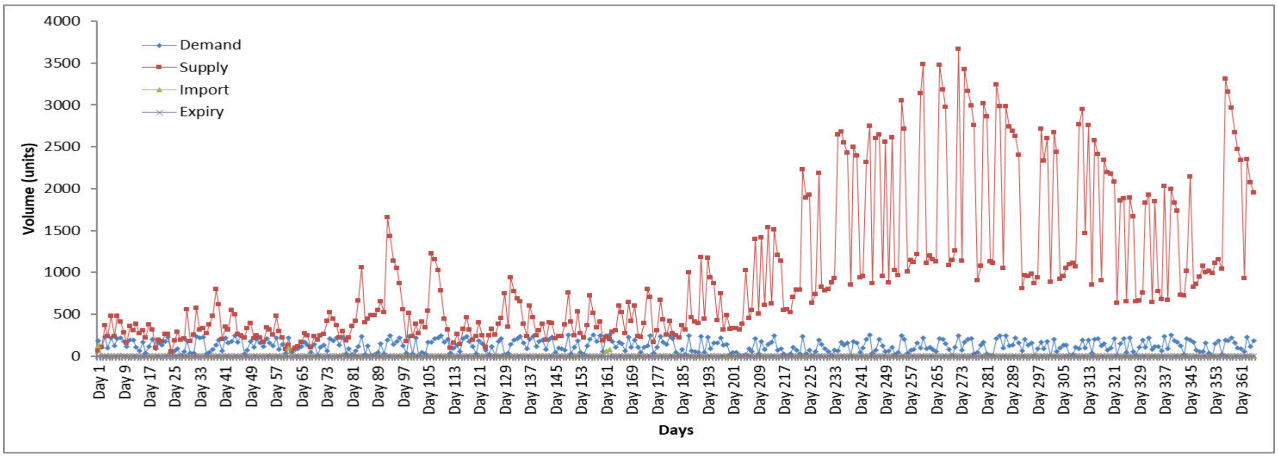

Figure 10: Representation of a line graph over a period of 365 days for the DA implementation of dataset 1

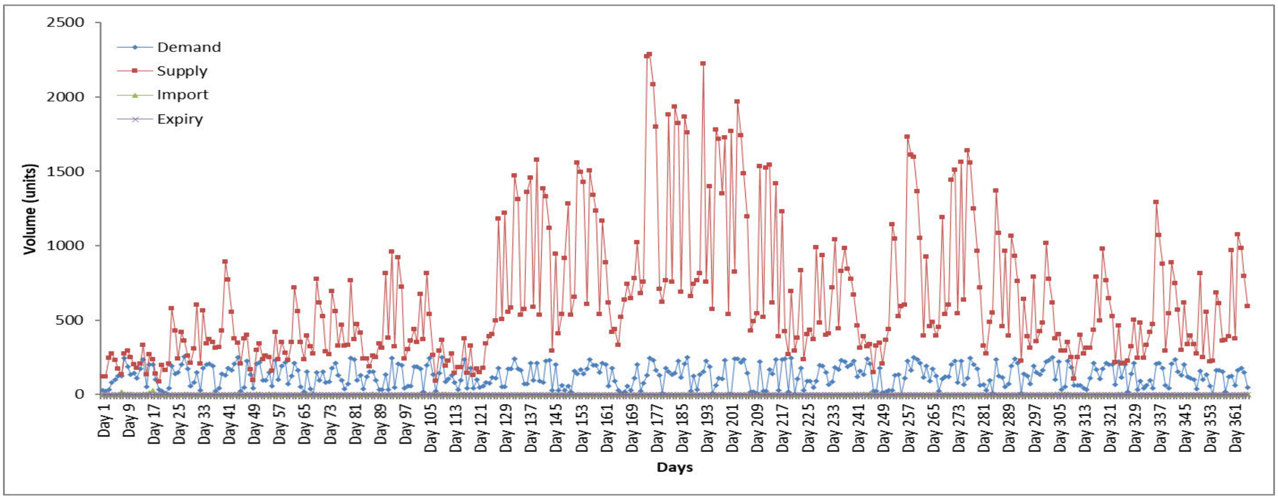

Figure 11: Representation of a line graph over a period of 365 days for the SOS implementation of dataset 1

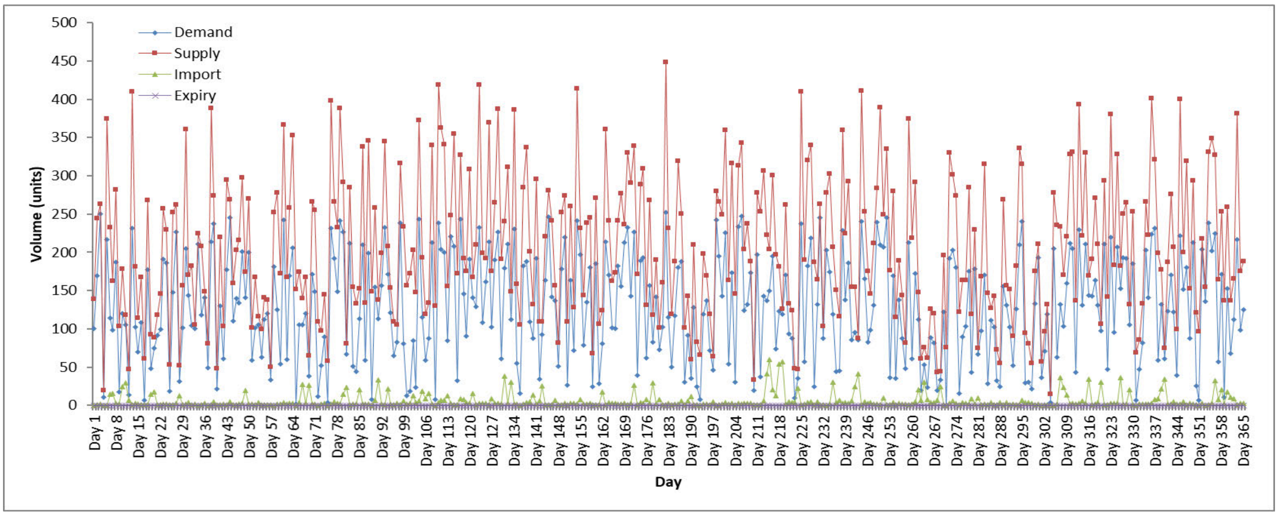

Figure 12: Representation of a line graph over a period of 365 days for the GWO implementation of dataset 1

4.2.2. Discussion for dataset 1

Figures 8-12 represent line graphs obtained when each metaheuristic was subjected to dataset 1. Table 9 illustrates the running times obtained per algorithm, most algorithms achieved a running time exceeding 70 minutes, however the PSO algorithm was much quicker achieving a duration time of 8.4 minutes. No algorithms experienced any form of expiry throughout the time frame:

- The GA algorithm experienced stock-piling around day 233, before this period imports occurred on sporadic days mainly for blood types B^+ and AB^+. Blood types A^+, A^- and O^- experienced no form of importation within the time frame. Around days 22, 85 and 127, the algorithm experienced very brief periods of stock-piling, however these lasted for periods of approximately 20 days.
- PSO experienced stock piling at a very early period (around day 90). However, low levels of importation still occurred throughout the 365-day period, with importation mainly being accredited to blood types A^+ and O^-. This implies that the algorithm struggled to stock-pile for these blood types in particular.
- DA experienced a brief period of stock-piling around day 73, however this only occurred for approximately 30 days. A steady rate of stock-piling only started to occur after day 180, this resulted in large amounts of WB units supply and no form of importation for any blood types.
- The SOS algorithm experienced stock-piling around day 120, with very little importation levels throughout the time frame.

- GWO produced random results throughout its time frame with no distinguishable pattern. The GWO algorithm never experienced stock-piling and had minimal importation amounts occurring frequently. The results also show that O^- blood was the only blood type not to experience importation, whilst blood types A^-, B^- and AB^- had relatively small importation averages. This implies that the algorithm was able to manage blood types that are rarer (scarce in society) more efficiently,

After assessing each algorithm, it was definitive that the SOS algorithm outperformed the other algorithms when subjected to dataset 1. Even though the SOS algorithm performed much slower (in terms of computational time) to the PSO system, it incurred the lowest averages relating to importation, and was the second fastest to experience stock-piling.

4.2.3 Dataset 2

Dataset 2 incorporates South African generated values in order to generate percentage bounds for each month (Table 5). Due to certain months having lower percentage bounds, the overall averages attained for both demand and supply are lower as compared dataset 1. A more in-depth comparison is discussed in Section 4.3

Table 10: Average results achieved for each metaheuristic implementation subjected to dataset 2 for each blood group measured in units.

MT	Variable	A^+	A^-	B^+	B^-	AB^+	AB^-	O^+	O^-
GA	Supply	157.87	26.43	59.48	20.51	13.08	9.35	192.73	39.42
	Demand	19.67	3.07	7.38	1.23	1.84	0.61	23.98	4.30
	Import	0.07	0.00	0.01	0.00	0.01	0.00	0.09	0.00
	Expiry	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
PSO	Supply	278.14	3.57	101.07	1.79	24.84	10.51	25.50	4.98
	Demand	19.28	3.01	7.23	1.20	1.81	0.60	23.50	4.22
	Import	1.29	0.71	0.09	0.31	0.23	0.04	5.60	0.96
	Expiry	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
DA	Supply	27.40	8.53	10.29	3.27	1.90	1.07	35.90	9.99
	Demand	20.01	3.13	7.51	1.25	1.88	0.63	24.39	4.38
	Import	2.61	0.09	0.81	0.03	0.60	0.12	1.38	0.13
	Expiry	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
SOS	Supply	222.87	43.48	76.14	24.06	19.19	9.34	294.67	52.15
	Demand	19.08	2.98	7.15	1.19	1.79	0.60	23.25	4.17
	Import	0.11	0.00	0.03	0.00	0.01	0.00	0.08	0.00
	Expiry	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
GWO	Supply	29.48	15.30	10.10	3.90	0.85	0.86	35.87	7.73
	Demand	19.96	3.12	7.48	1.25	1.87	0.62	24.33	4.37
	Import	0.72	0.01	0.52	0.05	1.02	0.17	0.57	0.00
	Expiry	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00

Table 11: Representation of the running time per metaheuristic for dataset 2

Metaheuristic	Time (Ms)	Time(Minutes)
GA	3983352	66.38
PSO	561452	9.35
DA	4113223	68.55
SOS	4665054	77.75
GWO	4525144	75.41

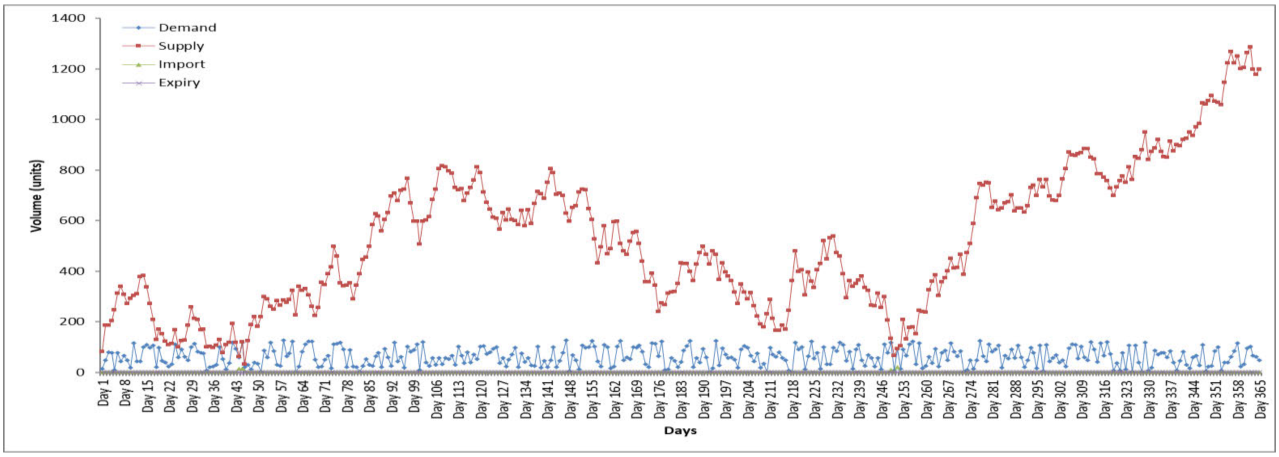

Figure 13: Representation a line graph over a period of 365 days for the GA implementation of dataset 2

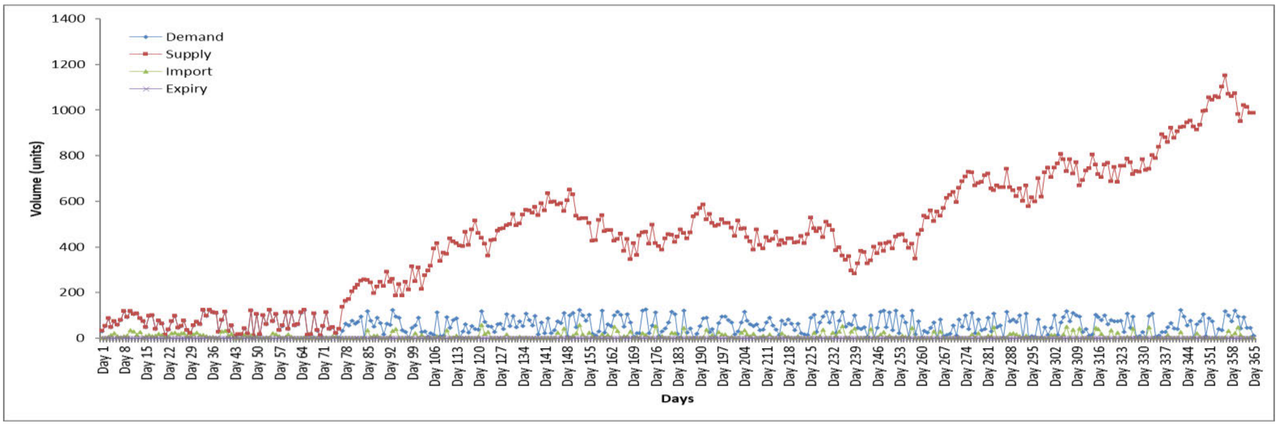

Figure 14: Representation of a line graph over a period of 365 days for the PSO implementation of dataset 2

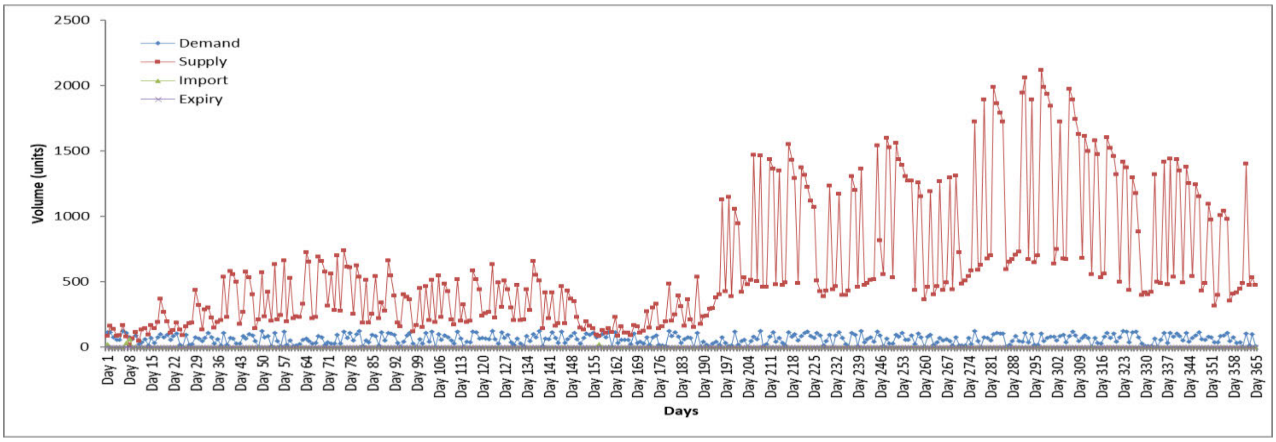

Figure 15: Representation of a line graph over a period of 365 days for the DA implementation of dataset 2

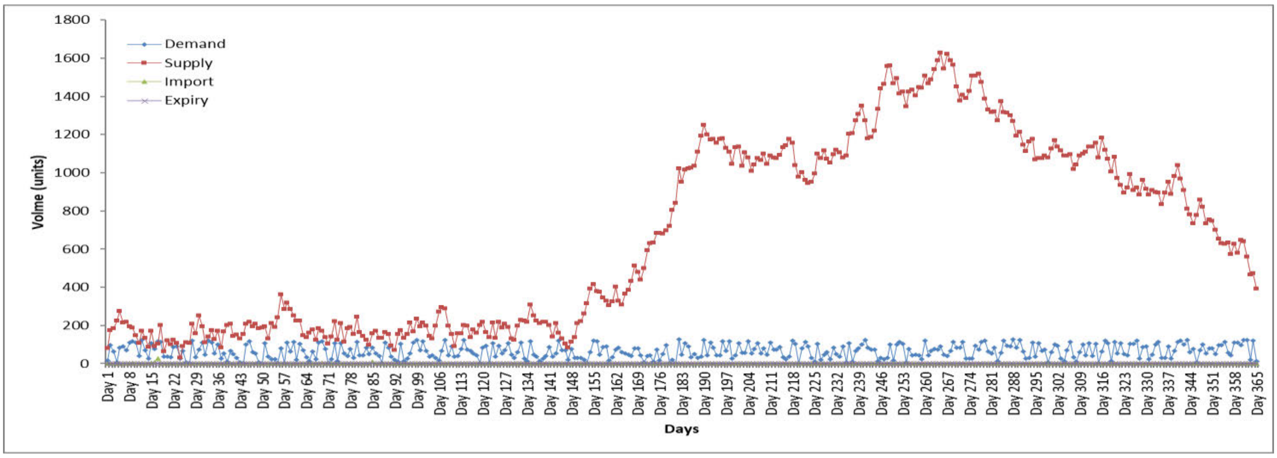

Figure 16: Representation of a line graph over a period of 365 days for the SOS implementation of dataset 2

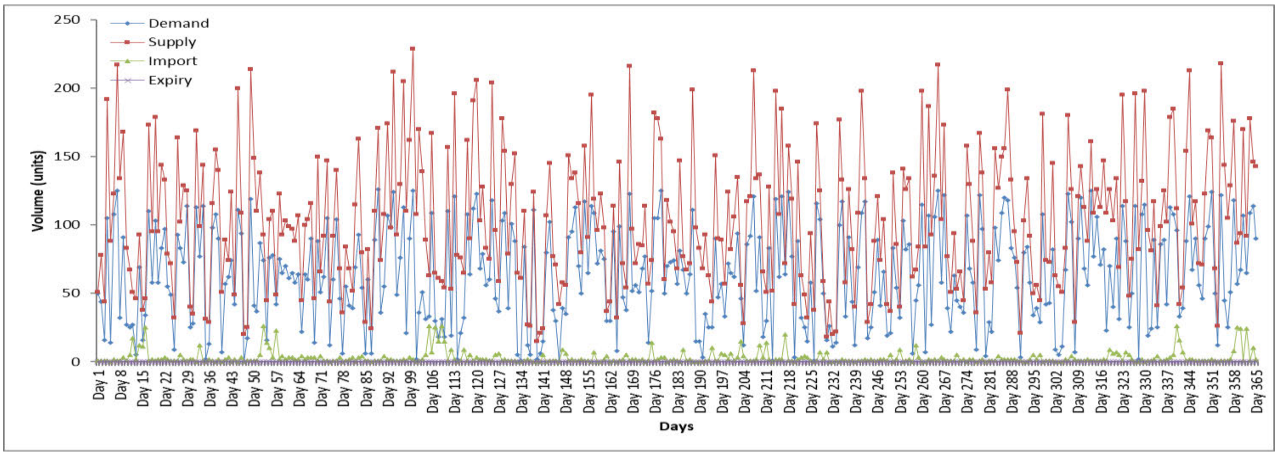

Figure 17: Representation of a line graph over a period of 365 days for the GWO implementation of dataset 2

4.2.4 Discussion for Dataset 2

Figures 13-17 represent line graphs obtained when each metaheuristic was subjected to Dataset 2, whilst Table 11 depicts the running times obtained per algorithm**.** In comparison to the running times achieved from Dataset 1, all the algorithms except the PSO achieved smaller computational times. A noteworthy aspect relates to the average demands attained per blood type. Due to the generation of demand values in Dataset 2 the average levels have decreased for the majority of the algorithms. Similar to Dataset 1, no form of expiry occurred for any of the algorithms:

- The GA algorithm experienced stock-piling at a very early period (around day 43), with the largest importation amounts occurring for blood types A^+ and O^+, implying that the algorithm cannot attain adequate blood supplies for two relatively abundant blood types.
- Similar to the results obtained from the GA algorithm, the PSO heavily imports for blood types A^+ and O^+. In addition, the PSO took a larger computational time in comparison to Dataset 1, and only achieved stock-piling around day 77.
- The line graph depicted in Figure 15 is similar to the pattern obtained from Dataset 1 results for DA, in that the supply for blood heavily fluctuates between days. Even though the DA algorithm achieved stock-piling at a very early stage (day 25), the accumulation of small importation levels for the less common blood types occurred throughout the time frame.
- The SOS algorithm achieved stock-piling around day 148 which in prospective??? to the other algorithms can be deemed as considerably slow. However, similar to dataset 1 the SOS algorithm achieved very low importation levels.
- In Dataset 1, the GWO algorithm did not achieve stock-piling, and there was no difference in Dataset 2. The average importation levels for GWO were relatively low, with importation occurring throughout the time frame.

Due to the algorithm experiencing stock-piling at an early period coupled with its low importation levels for the certain blood types, the GA can be deemed as the more efficient algorithm for Dataset 2. Dataset 2 tried to emphasize the concept of generating data based on South African statistics, the fluctuating demand bounds directly link with the decrease in total averages attained per blood type as compared to Dataset 1.

4.2.5 Dataset 3

Dataset 3 served to examine how well each metaheuristic algorithm performed when subjected to an instance of the demand exceeding the supply. As expected the average levels of demand per blood type exceeded the average supply. This is a scenario which was identified as an expense to the blood bank due to additional WB units needing to be imported to satisfy the demand in a day.

Table 12: Average results achieved for each metaheuristic implementation subjected to Dataset 3 for each blood group measured in units.

MT	Variable	A^+	A^-	B^+	B^-	AB^+	AB^-	O^+	O^-
GA	**Supply**	25.72	4.12	9.65	1.67	2.32	0.90	31.31	5.68
	Demand	38.69	6.05	14.51	2.42	3.63	1.21	47.16	8.46
	Import	18.96	2.94	7.11	1.18	1.82	0.61	23.11	4.12
	Expiry	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
PSO	**Supply**	33.26	8.13	16.11	5.07	0.97	2.54	47.98	12.35
	Demand	40.54	6.33	15.20	2.53	3.80	1.27	49.41	8.87
	Import	7.69	0.40	1.19	0.08	2.86	0.24	3.05	0.13
	Expiry	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
DA	**Supply**	41.47	6.48	15.55	2.59	3.89	1.30	50.55	9.07
	Demand	48.83	16.24	17.84	6.15	2.19	1.73	56.38	18.24
	Import	7.07	0.68	2.85	0.28	2.06	0.26	8.13	0.91
	Expiry	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
SOS	**Supply**	68.90	28.51	27.04	11.63	3.08	2.40	72.98	30.71
	Demand	37.16	5.81	13.93	2.32	3.48	1.16	45.29	8.13
	Import	1.20	0.01	0.39	0.01	1.51	0.10	0.68	0.00
	Expiry	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
GWO	**Supply**	16.14	3.20	6.25	0.38	0.85	0.05	19.01	4.46
	Demand	21.06	3.29	7.90	1.32	1.97	0.66	25.66	4.61
	Import	6.43	1.14	2.34	1.10	1.15	0.63	8.10	0.97
	Expiry	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00

Table 13: Running time per metaheuristic for Dataset 3

Metaheuristic	Time (Ms)	Time(Minutes)
GA	3872251	64.54
PSO	494639	8.24
DA	4203022	70.05
SOS	4765186	79.42
GWO	4623043	77.05

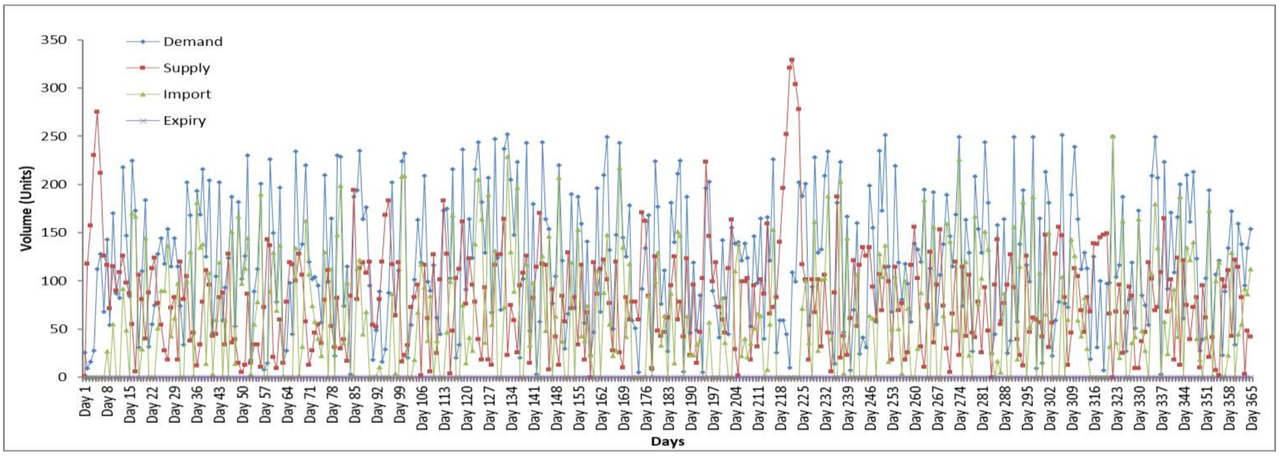

Figure 18: Representation of a line graph over a period of 365 days for the GA implementation of Dataset 3

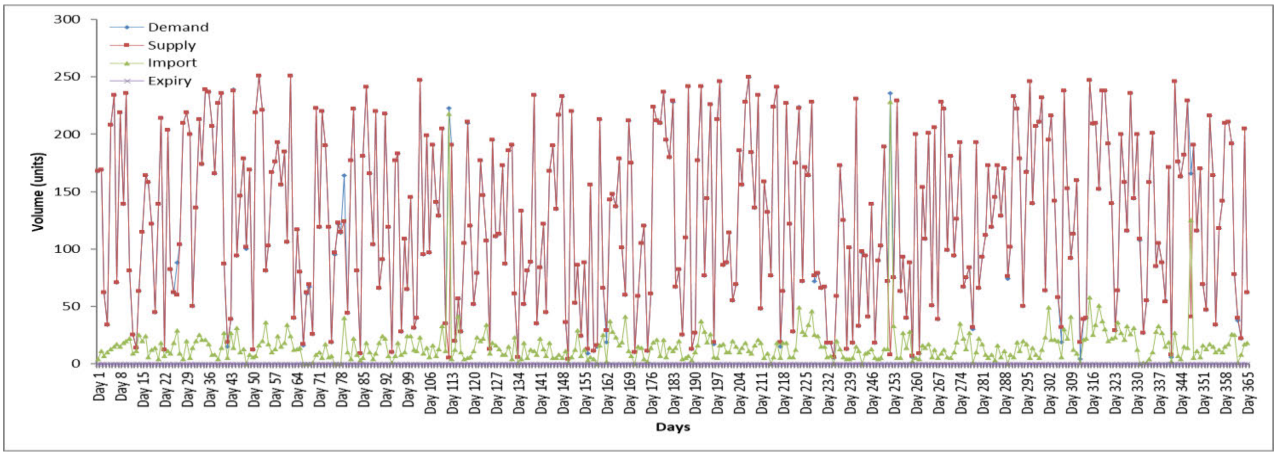

Figure 19: Representation of a line graph over a period of 365 days for the PSO implementation of Dataset 3

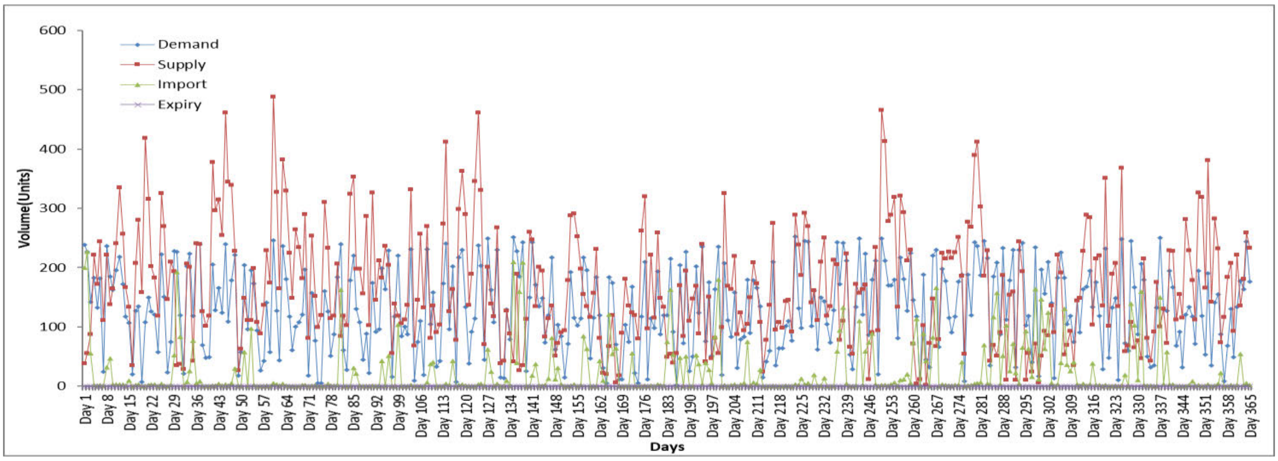

Figure 20: Representation of a line graph over a period of 365 days for the DA implementation of Dataset 3

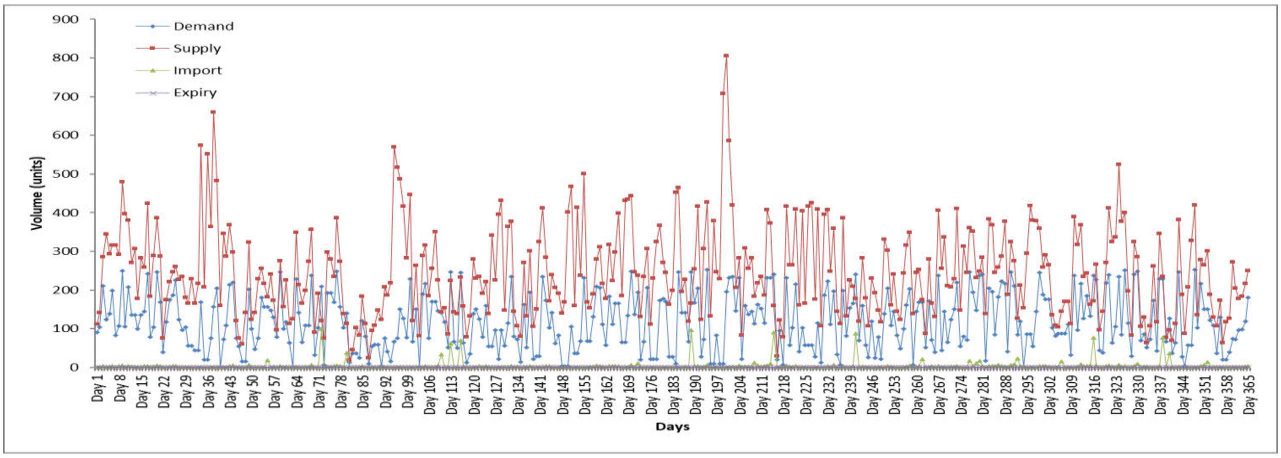

Figure 21: Representation of a line graph over a period of 365 days for the SOS implementation of Dataset 3

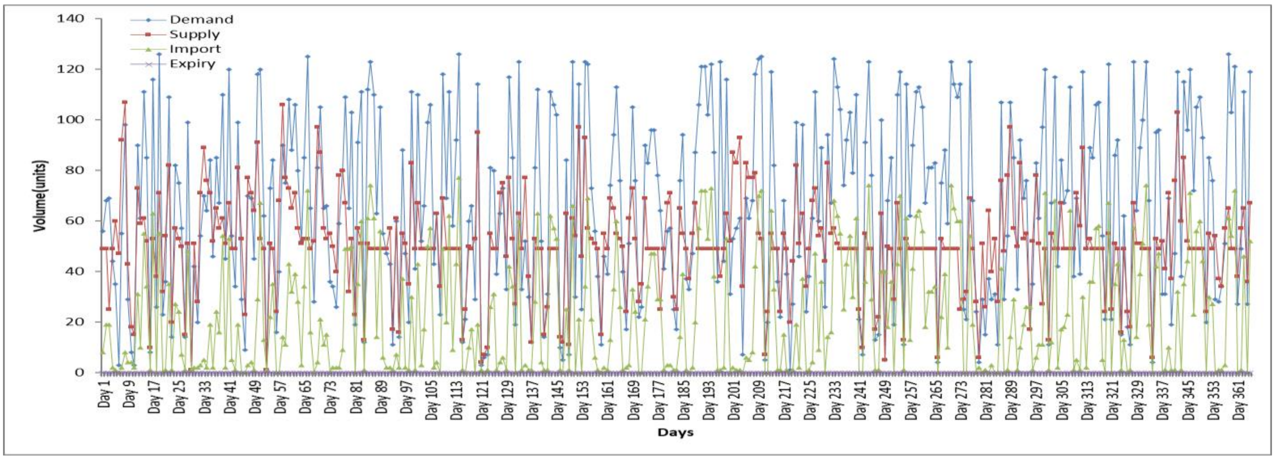

Figure 22: Representation of a line graph over a period of 365 days for the GWO implementation of Dataset 3

4.2.6. Discussion for Dataset 3

Dataset 3 tested how well a blood bank coped when the demand exceeded the supply on hand for any given day. None of the algorithms achieved stock-piling, and achieved drastic increases in regards to the average importation levels for each blood type. Figures 18-22 represent line graphs obtained when each metaheuristic was subjected to Dataset 2, whilst Table 13 depicts the running times obtained per algorithm**.** In comparison to the running times achieved from Dataset 1, all algorithms experienced a decrease in total time taken for the algorithm to terminate. There were no significant aspects to analyse per algorithm as no stock-piling or expiry occurred. Overall the SOS system delivered far fewer importation levels per blood type and can be deemed as the best algorithm to work Dataset 3.

4.2.7 Dataset 4

Dataset 4 tested the opposite of Dataset 3 whereas the supply exceeded the demand within the time frame. The level of expiry was expected to increase, however due to the lifespan of a WB unit being 30 days the level of expiry across any algorithm remained at 0. If expiry did occur, it would have implied that the metaheuristic algorithm could not utilise WB unit resources efficiently, and in the real-world blood banks would be expected to dispose of these unusable WB units in a proper manner which incurs additional expenses.

Table 14: Average results achieved for each metaheuristic implementation subjected to Dataset 3 for each blood group measured in units

MT	Variable	A+	A-	B+	B-	AB+	AB-	O+	O-
GA	**Supply**	2124.89	335.79	791.17	152.07	195.07	66.28	2583.42	469.82
	Demand	20.62	3.22	7.73	1.29	1.93	0.64	25.13	4.51
	Import	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
	Expiry	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
PSO	**Supply**	3584.50	6.29	1350.75	2.52	328.10	100.46	49.15	8.80
	Demand	20.65	3.23	7.74	1.29	1.94	0.65	25.17	4.52
	Import	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
	Expiry	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
DA	**Supply**	1893.13	470.49	715.33	169.36	175.23	56.17	506.27	461.14
	Demand	19.71	3.08	7.39	1.23	1.85	0.62	24.02	4.31
	Import	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
	Expiry	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
SOS	**Supply**	3419.58	565.67	1308.53	209.98	326.20	98.90	4172.27	746.18
	Demand	18.84	2.94	7.07	1.18	1.77	0.59	22.96	4.12
	Import	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
	Expiry	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
GWO	**Supply**	3977.10	617.69	1498.53	257.23	363.27	118.53	4843.29	868.76
	Demand	20.31	3.17	7.61	1.27	1.90	0.63	24.75	4.44
	Import	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
	Expiry	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00

Table 15: Running time per metaheuristic for dataset 4

Metaheuristic	Time (Ms)	Time(Minutes)
GA	5220157	87.00
PSO	500968	8.34
DA	5135744	85.60
SOS	5978164	99.63
GWO	5998772	99.97

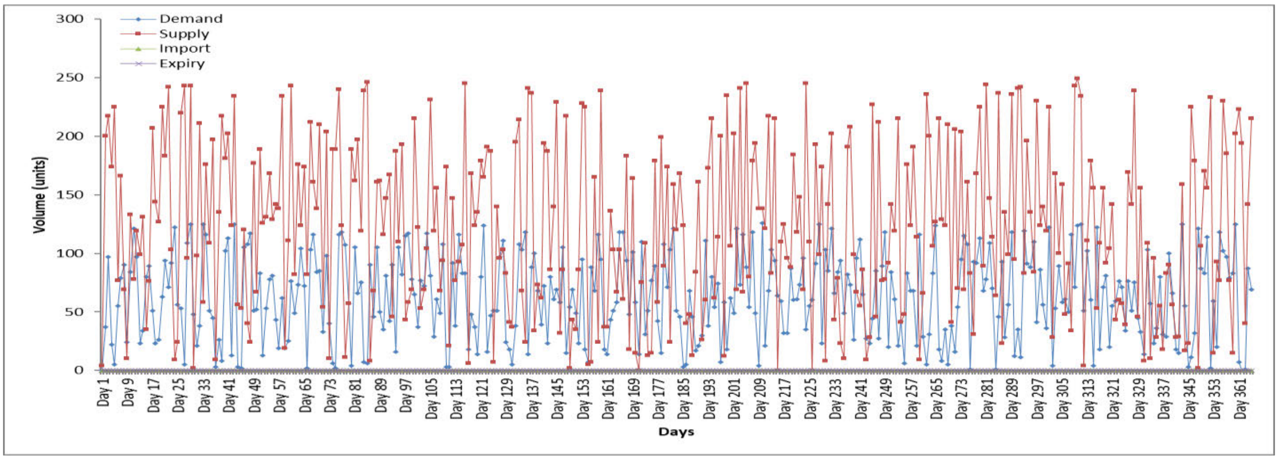

Figure 23: Representation of a line graph over a period of 365 days for the GA implementation of Dataset 4

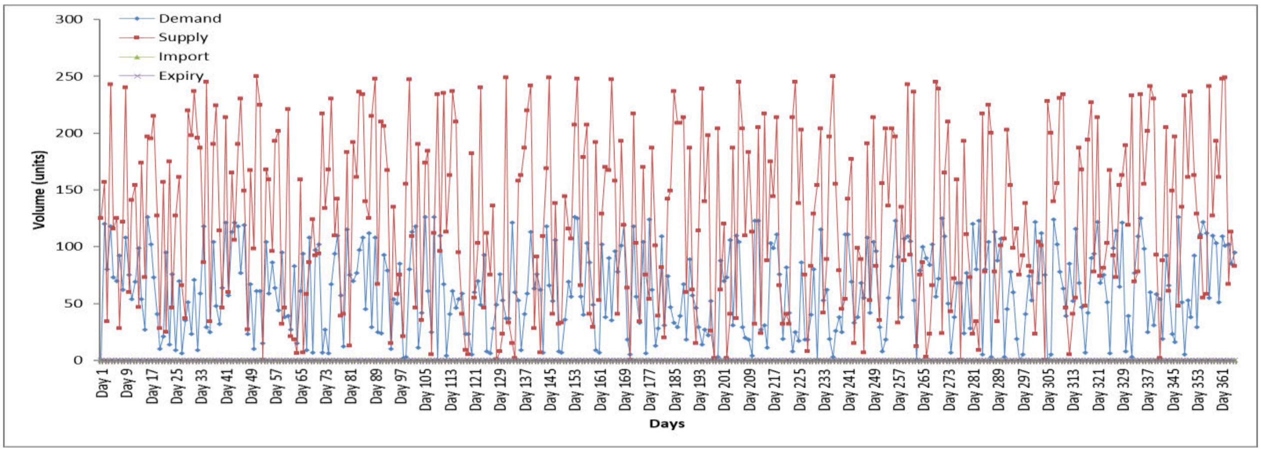

Figure 24: Representation of a line graph over a period of 365 days for the PSO implementation of Dataset 4

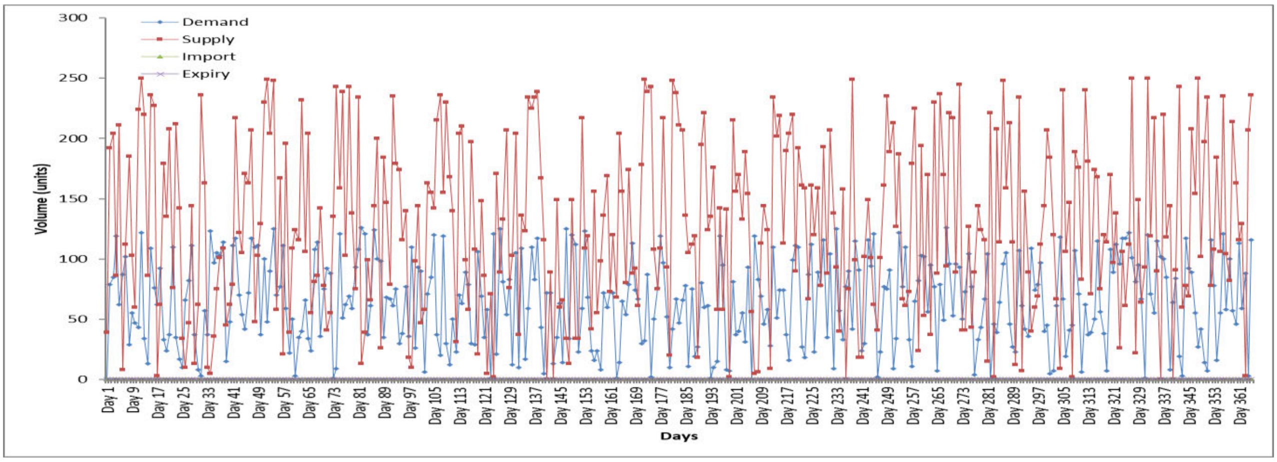

Figure 25: Representation of a line graph over a period of 365 days for the PSO implementation of Dataset 4

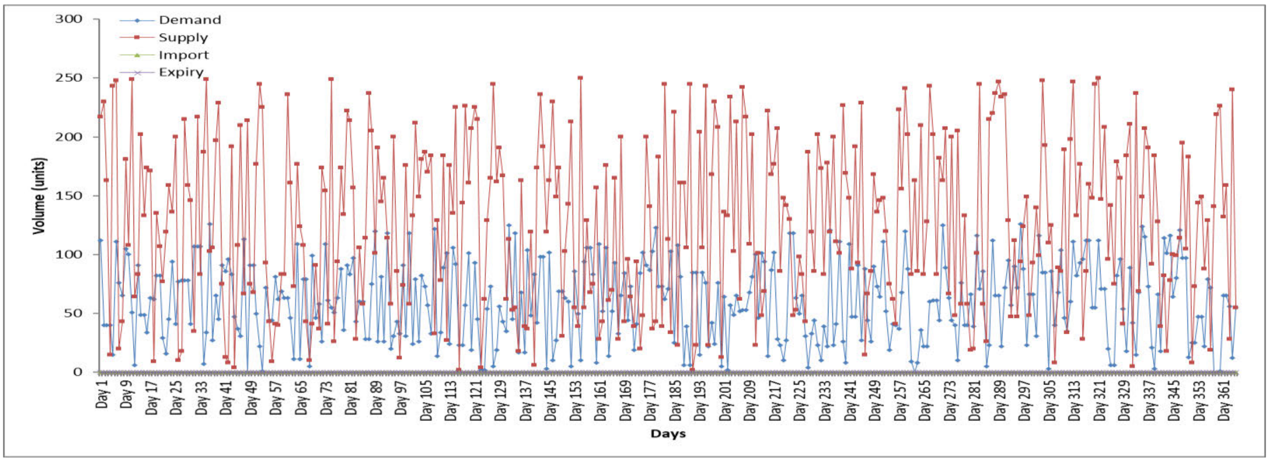

Figure 26: Representation of a line graph over a period of 365 days for the SOS implementation of Dataset 4

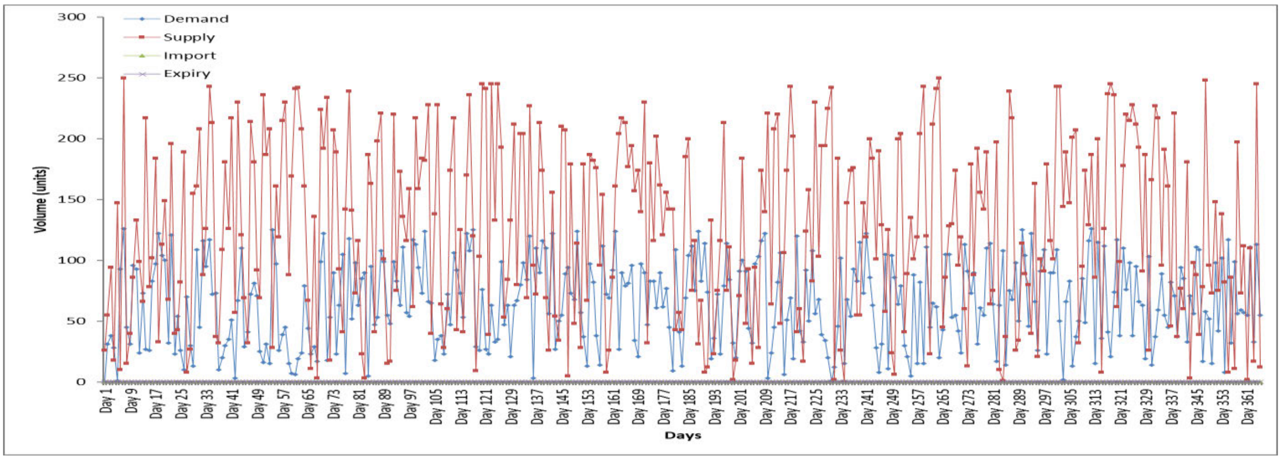

Figure 27: Representation of a line graph over a period of 365 days for the GWO implementation of Dataset 4

4.2.8. Discussion for dataset 4

Dataset 4 tested the opposite of Dataset 3, as the percentage bounds for generating supply greatly outweighed the bounds for generating demand. In theory, the levels of expiry were expected to drastically increase. However, after analysing the results, no form of expiry occurred due to the 30 lifespan of WB units. Practically it is unrealistic for a WB unit to stay in storage above its allotted lifespan unless demand is non-existent. Dataset 4 revealed that all the algorithms shared similar graphical shapes with the supply trend exceeding the demand trend. In addition, none of the algorithms experienced importation or expiration with similar computational times as compared to Dataset 1. Taking these factors into account, all algorithms performed efficiently when subjected to Dataset 4.

4.2.9. Dataset 5

A blood bank can be subjected to a mass collection of WB units, which is what Dataset 5 examined. The percentage bounds used for demand and supply replicated the bounds used in Dataset 1, but utilised 5000 WB units as an initial volume instead of 500 units.

Table 16: Average results achieved for each metaheuristic implementation subjected to Dataset 5 for each blood group measured in units.

MT	Variable	A^+	A^-	B^+	B^-	AB^+	AB^-	O^+	O^-
GA	**Supply**	5205.15	816.14	1952.03	320.22	485.21	159.66	6334.86	1136.78
	Demand	383.20	59.87	143.70	23.95	35.92	11.97	467.02	83.82
	Import	1.88	0.31	0.70	0.10	0.28	0.06	2.31	0.39
	Expiry	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
PSO	**Supply**	3331.25	80.35	1264.40	40.95	307.45	106.53	503.23	108.92
	Demand	397.69	62.14	149.13	24.86	37.28	12.43	484.68	86.99
	Import	46.06	14.62	5.68	5.70	6.45	0.40	129.94	19.75
	Expiry	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
DA	**Supply**	3442.04	1749.02	1325.24	674.90	310.58	108.34	2190.95	1773.78
	Demand	396.28	61.92	148.60	24.77	37.15	12.38	482.97	86.69
	Import	2.87	0.15	0.57	0.00	1.91	0.03	2.24	0.09
	Expiry	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
SOS	**Supply**	2404.66	1184.73	882.15	438.40	238.29	86.86	1639.28	1220.06
	Demand	397.86	62.17	149.20	24.87	37.30	12.43	484.89	87.03
	Import	0.00	0.00	0.00	0.00	0.62	0.00	0.00	0.00
	Expiry	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
GWO	**Supply**	637.34	252.61	230.45	70.28	19.65	17.99	771.21	161.06
	Demand	424.51	66.33	159.19	26.53	39.80	13.27	517.38	92.86
	Import	7.73	0.21	6.15	0.39	20.15	0.93	5.22	0.00
	Expiry	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00

Table 17: Running time per metaheuristic for Dataset 5

Metaheuristic	Time (Ms)	Time(Minutes)
GA	4422080	73.70
PSO	504499	8.40
DA	4203022	70.05
SOS	5805914	96.76
GWO	4385240	73.09

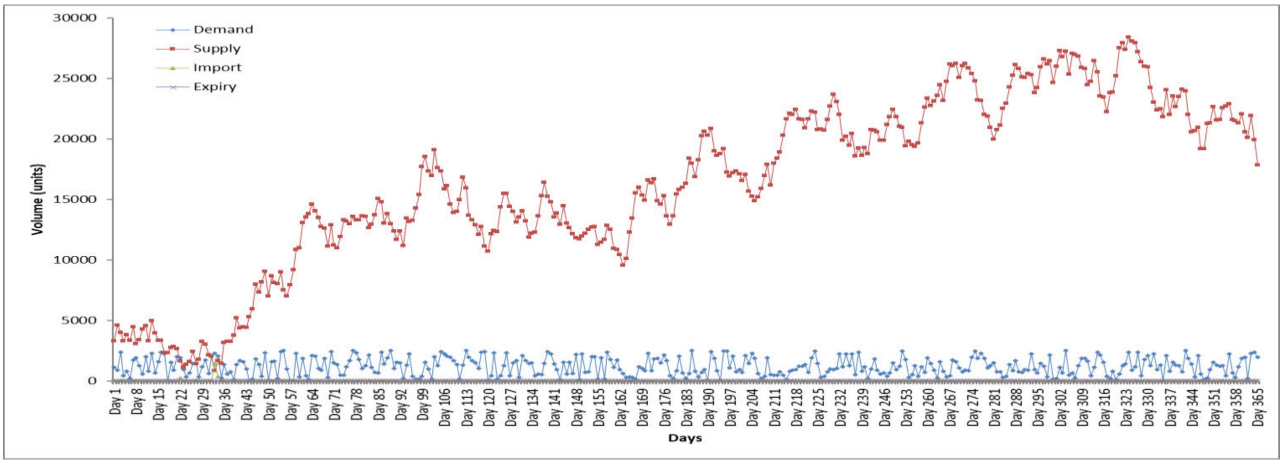

Figure 28: Representation of a line graph over a period of 365 days for the GA implementation of Dataset 5

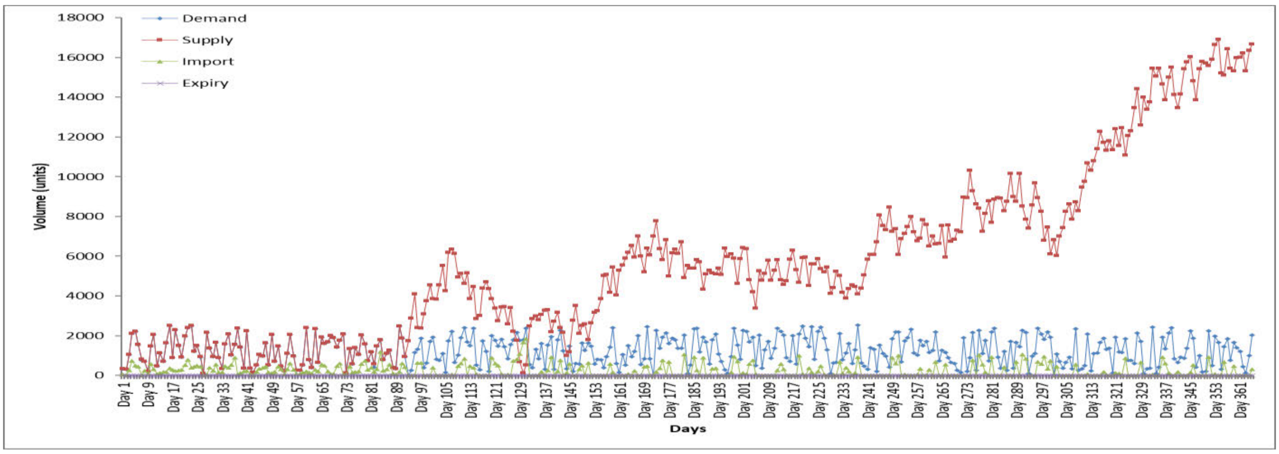

Figure 29: Representation of a line graph over a period of 365 days for the PSO implementation of Dataset 5

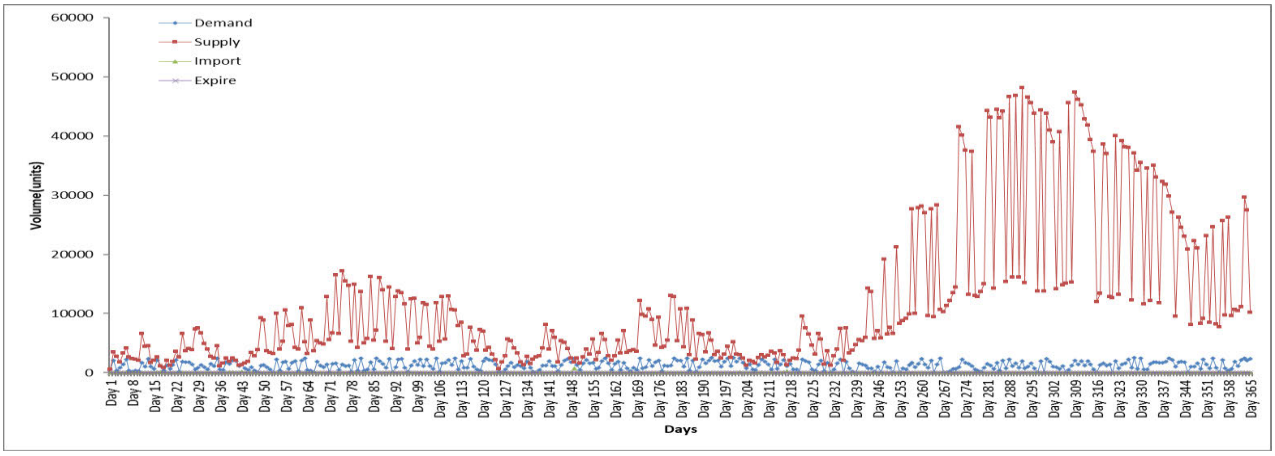

Figure 30: Representation of a line graph over a period of 365 days for the DA implementation of Dataset 5

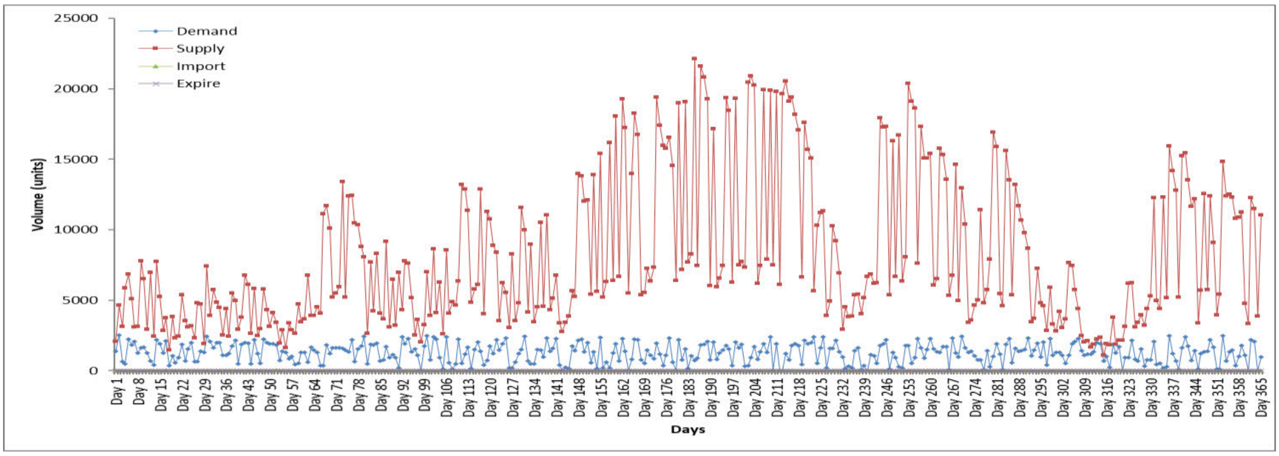

Figure 31: Representation of a line graph over a period of 365 days for the SOS implementation of Dataset 5

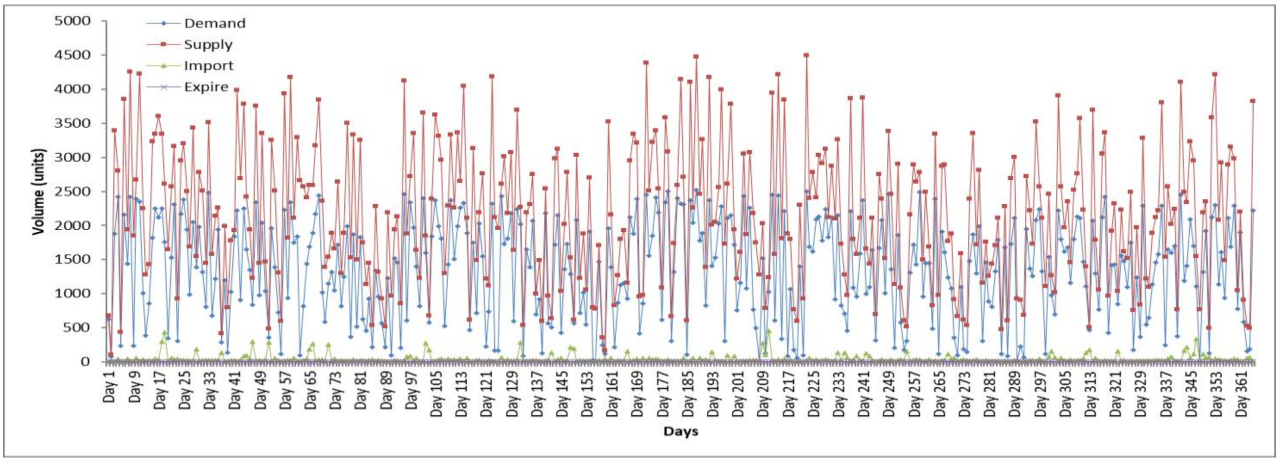

Figure 32: Representation of a line graph over a period of 365 days for the GWO implementation of Dataset 5

4.2.10. Discussion of dataset 5

Dataset 5 was a replica of Dataset 1 in the aspect of percentage bounds for demand and supply. The only difference lay in the initial volume of blood units which was 5000 units. This tested how well a blood bank was able to efficiently distribute large volumes of WB units. As expected the average levels for demand and supply increased accordingly. The increase in WB volume had very little effect on the computational time for the algorithms, and no form of expiry occurred:

- GA in proportion to the large volume of WB units, experience very low amounts of importation, and similar to Dataset 1 experienced stock-piling at a very early stage (approximately day 35). After stock-piling, there were no visible forms of importation experienced throughout the remainder of the time frame.
- PSO delivered very high importation levels especially for blood types A^+ and O^+ with stock-piling occurring around day 150. Similar to Dataset 1, the PSO algorithm still imported WB units even after stock-piling occurred.
- The DA was similar to the GA in that the importation levels were quite low with blood type B^- experiencing no form of imports. Stock-piling occurred around day 240 after which the supply trends followed the same fluctuating pattern as compared to dataset 1.
- SOS system posed the largest change in comparison to Dataset 1. The algorithm experienced stock-piling from day 1 which resulted in no form of importation except for blood type AB^+.
- GWO followed its typical sporadic graphical trend with no stock-piling occurring within the timeframe, as such the GWO implementation incurred imports for specific blood types throughout the allotted time frame.

Dataset 1 used an initial volume of 500 WB units, whilst Dataset 5 used 5000 units. Even though there was a drastic difference between the initial blood unit volumes, most algorithms displayed similar behaviour and graphical trends. The SOS was by far the most effective algorithm with almost no form of importation and was the fastest to reach the effect of stock-piling.

4.2.11 Dataset 6

Since this study's aim relates to incorporating SAGV for stochastically generating demand and supply values, the introduction of Dataset 6 seemed imperative. Dataset 6 follows the same demand bounds of Dataset 2, but utilised a higher initial WB unit volume of 5000 units.

Table 18: Average results achieved for each metaheuristic implementation subjected to Dataset 6 for each blood group measured in units

MT	Variable	A^+	A^-	B^+	B^-	AB^+	AB^-	O^+	O^-
GA	**Supply**	17293.71	2700.27	6485.94	1082.13	1619.56	536.04	21076.53	3779.62
	Demand	345.93	54.05	129.72	21.62	32.43	10.81	421.60	75.67
	Import	0.73	0.11	0.28	0.05	0.07	0.03	0.90	0.16
	Expiry	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
PSO	**Supply**	10808.48	59.98	4047.42	24.02	1011.76	334.79	467.84	83.99
	Demand	351.56	54.93	131.84	21.97	32.96	10.99	428.47	76.90
	Import	0.29	17.44	0.11	6.96	0.03	0.01	136.02	24.41
	Expiry	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
DA	**Supply**	15915.12	7070.80	5962.89	2671.17	1466.52	489.37	7503.44	7096.10
	Demand	344.48	53.82	129.18	21.53	32.29	10.76	419.83	75.35
	Import	0.90	0.00	0.24	0.00	0.29	0.00	0.88	0.00
	Expiry	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
SOS	**Supply**	5157.35	914.22	2439.90	430.83	549.99	206.36	7621.29	1389.61
	Demand	354.62	55.41	132.98	22.16	33.25	11.08	432.20	77.57
	Import	0.00	0.00	0.00	0.00	0.18	0.00	0.00	0.00
	Expiry	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
GWO	**Supply**	830.88	360.82	292.25	106.79	14.26	23.18	963.91	205.18
	Demand	349.75	54.65	131.16	21.86	32.79	10.93	426.26	76.51
	Import	4.83	0.04	7.61	0.35	38.30	1.14	7.45	0.00
	Expiry	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00

Table 19: Running time per metaheuristic for Dataset 6

Metaheuristic	Time (Ms)	Time(Minutes)
GA	4321070	72.017
PSO	503324	8.38
DA	4302052	71.70
SOS	5906312	98.43
GWO	4362574	72.70

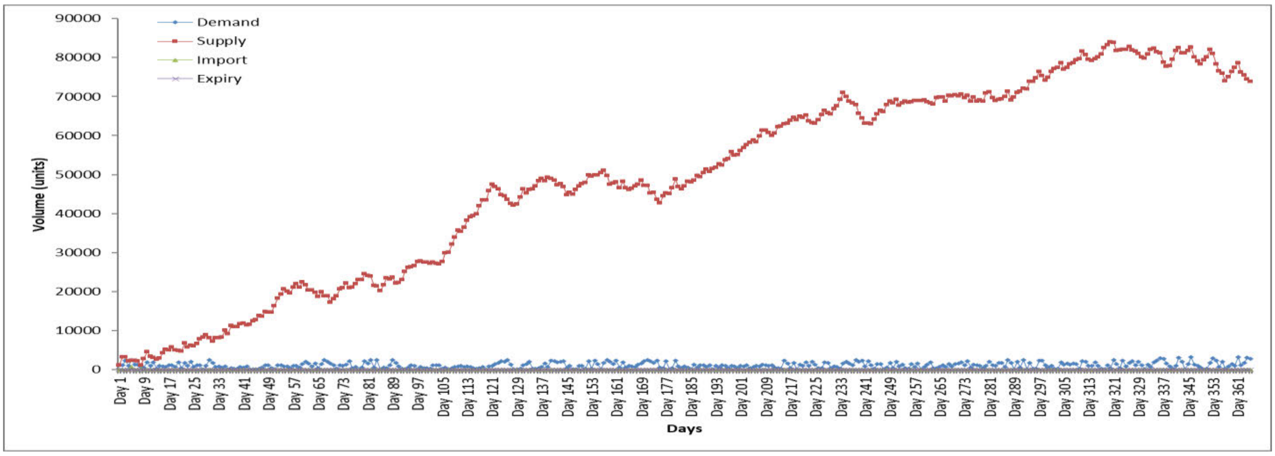

Figure 33: Representation of a line graph over a period of 365 days for the GA implementation of Dataset 6

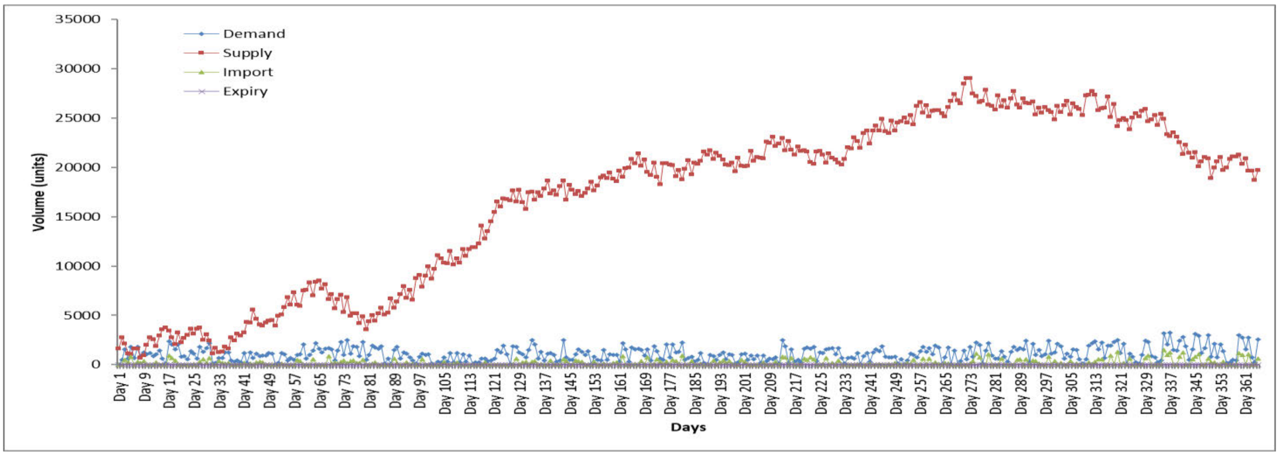

Figure 34: Representation of a line graph over a period of 365 days for the PSO implementation of Dataset 6

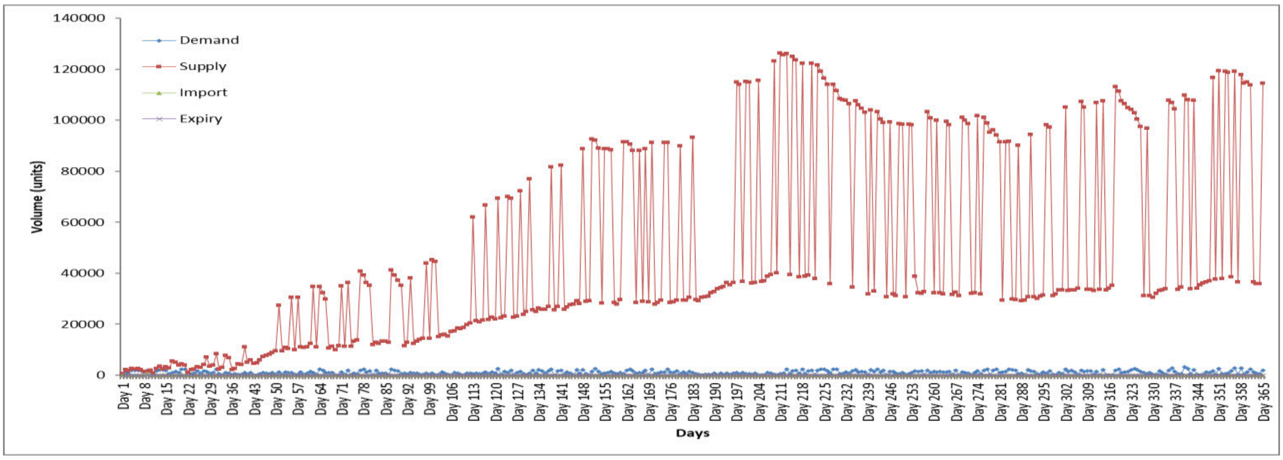

Figure 355: Representation of a line graph over a period of 365 days for the DA implementation of Dataset 6

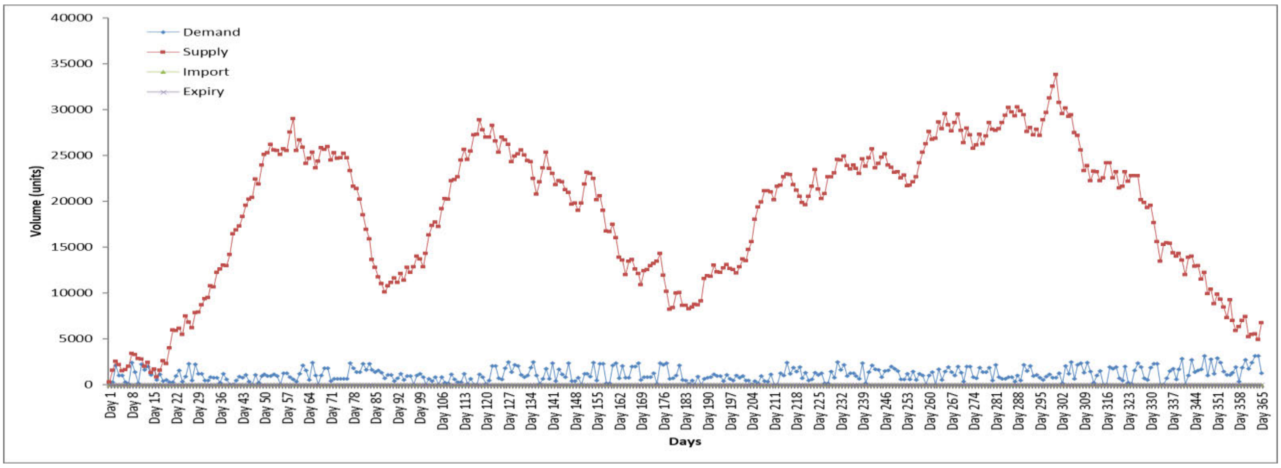

Figure 36: Representation of a line graph over a period of 365 days for the SOS implementation of Dataset 6

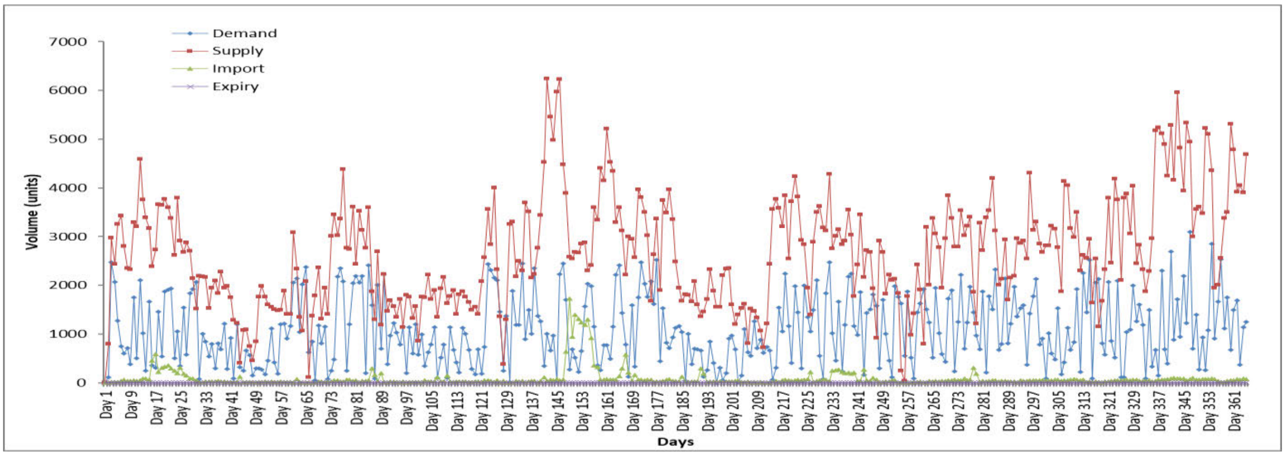

Figure 37: Representation of a line graph over a period of 365 days for the GWO implementation of Dataset 6

4.2.12. Discussion of Dataset 6

The last analysis of results related to Dataset 6. Dataset 6 was a replica of Dataset 2 in terms of percentage bounds, but was similar to Dataset 5 in testing a larger volume of WB units. Due to the percentage bounds being generated by South African statistics on a monthly basis, the demand and supply averages were expected to decrease. The results are as follows:

- GA performed efficiently, with stock-piling occurring around day 14, but did experience relatively high importation amounts before stockpiling occurred. Overall the average importation levels for each blood type are relatively small in comparison to the initial volume of WB units.
- The results obtained from the PSO algorithm was a large improvement than the previous Dataset 5 results. However, blood type O^+ still experienced very large importation levels throughout the time period. The algorithm did experience stock piling around day 35, but similar to the previous datasets sporadic amounts of importation still occurred even though the event of stock-piling occurred.
- For the DA implementation, blood types A^+, B^+, AB^+ and O^+ were the only types to experience importation, however these values where relatively small. The algorithm experienced stock-piling around day 40, and followed its fluctuating supply trend thereafter.
- Unlike Dataset 5, the SOS algorithm did not experience stockpiling from the day 1, but rather around day 17. After stock-piling occurred, no form of importation was experienced. Similar to Dataset 5, the only blood type to experience importation was type AB^+
- In previous datasets, the GWO did not experience any form of stock-piling. However, Dataset 6 reveals small periods of stock-piling which could be due to the unique nature of WB unit demand generation. This was an improvement as stock-piling decreased the overall importation experienced for each blood type.

Even though the SOS algorithm was the second fastest to experience stock-piling, it was still considered as the best algorithm for Dataset 6. SOS experienced no form of importation, except for blood type AB^+

4.3. Comparison between demand generations

As emphasized in the previous chapters, this study incorporated a method for generating values for WB unit demands by incorporating statistics from South Africa. Previous literature used fixed percentage bounds over a set time frame, which was illustrated in this study by Dataset 1. As proven by the results analysed in Dataset 1 and 2, the average demand and supply per blood type in Dataset 2 was smaller due to unique percentage bounds allocated to each month instead of a constant percentage bound used in Dataset 1. The statistics used to generate Dataset 2 includes public holidays and breaks from educational institutions within South Africa. The idea behind using such statistics is to emphasize the fact that WB units would have a higher demand in months which experienced an increase in dangerous activities such as drinking and driving, criminal events, etc. Therefore, this section serves as an analysis between the results obtained from generating demand values between the two mentioned methods. Below are 5 line graphs (figures 38–42) depicting the curvature of demand generation using the fixed percentage bound in comparison to the SAGV method per metaheuristic algorithm.

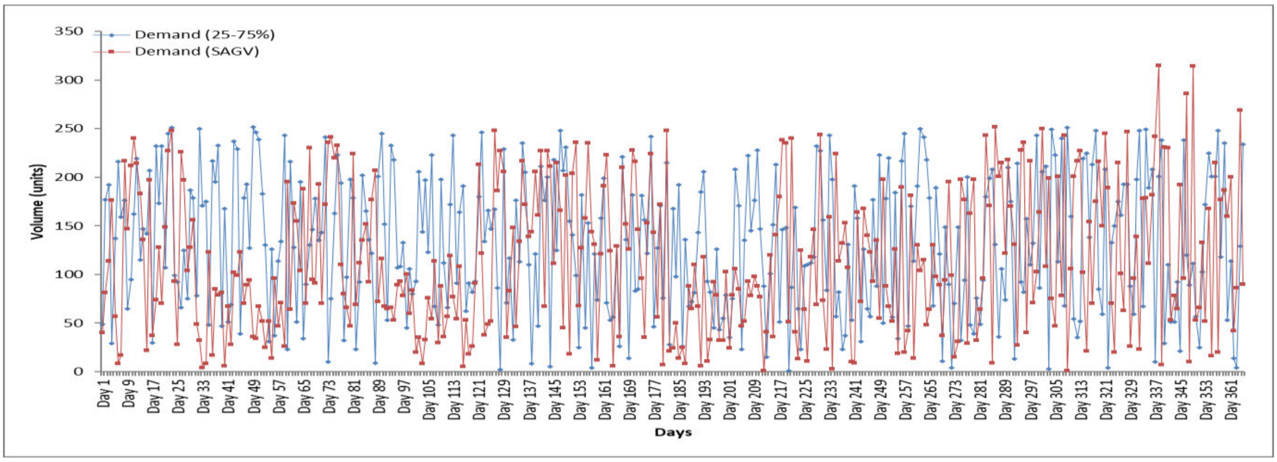

Figure 38: Representation of a comparison between demand generations for GA

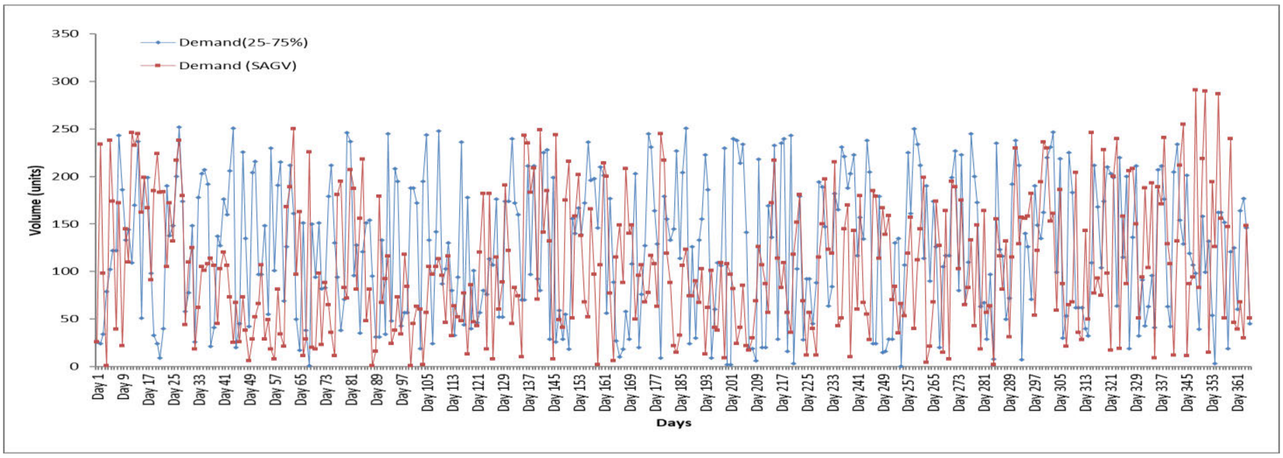

Figure 39: Representation of a comparison between demand generations for PSO

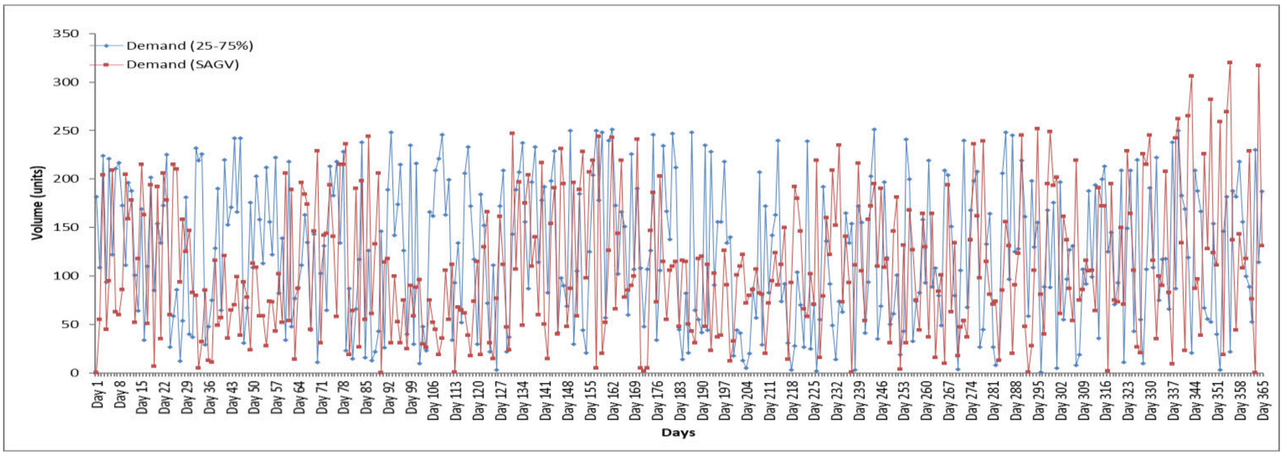

Figure 40: Representation of a comparison between demand generations for DA

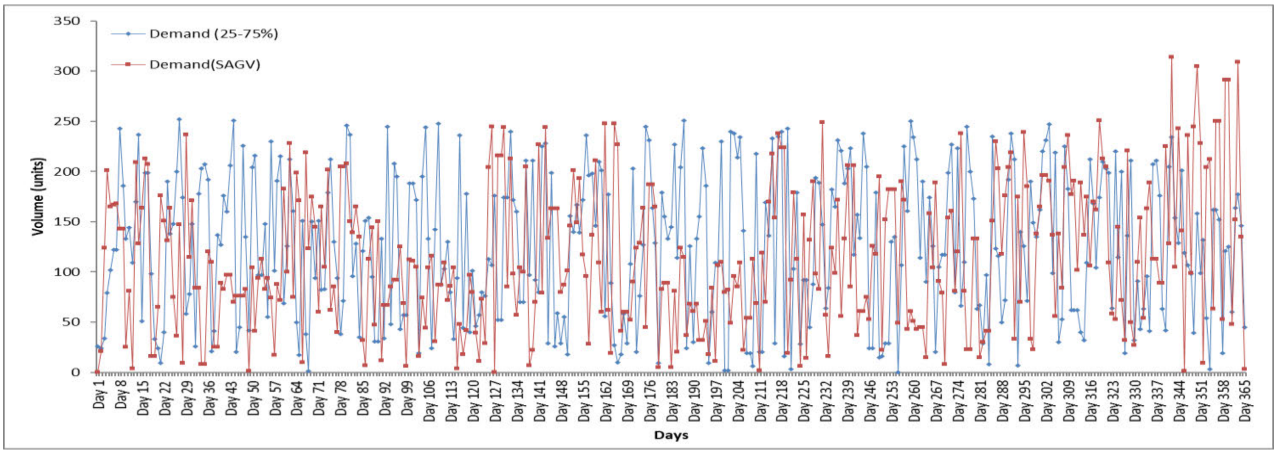

Figure 41: Representation of a comparison between demand generations for SOS

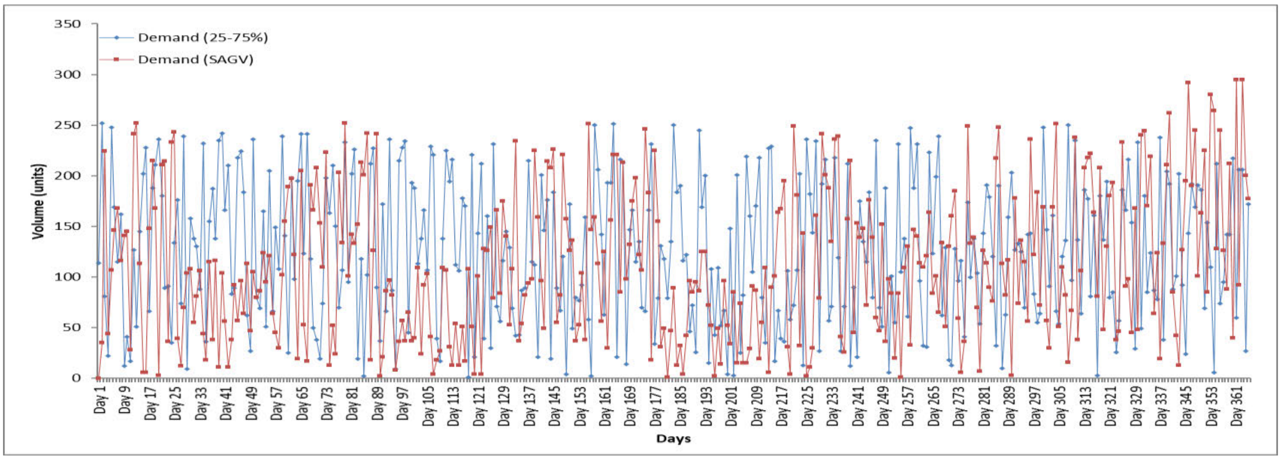

Figure 42: Representation of a comparison between demand generations for GWO

Figures 38-42 represent line graphs depicting the demand generated using fixed percentage bounds versus bounds generated using SAGV. A noticeable component relating to a fixed percentage bound relates to the absolute randomness in regards to the line trend, whereas the trend based on SAGV seems to have periods with high and low variations. With reference to SAGV, around days 28-55 (represents February) across Figures 38–42 indicate that the demand for WB units is generally lower, which correlates to the assumption that months with fewer public holidays and breaks from educational institutions should utilise fewer WB units. The opposite assumption relates to high WB unit demand which is clearly depicted in a month like December (days 331-365) which illustrates a much higher demand curvature.

4.4 Comparison with results from the literature

Previous research pertaining to the BAP utilised different blood banking policies and metaheuristics in comparison to this study. To assess the results obtained in this study, comparisons were made with regards to the average amount of importation and expiration. Due to the differences in mathematical models across some of the literatures, a precise comparison could not be attained. Taking this factor into account, it was possible to examine certain results achieved from studies conducted in [3, 7, 28]. Each of these studies implemented different metaheuristic algorithms with very similar blood banking structures, and identified the best algorithm that produced satisfactory results. Table 20 illustrates the best algorithms in accordance to each individuals study.

Table 20: Representation of the best algorithms in association to each research contribution towards the BAP

Reference	Best metaheuristic algorithm
[28]	GRASP
[7]	PSO
[3]	HC

The study conducted by [28] recorded cumulative amounts correlating to importation for datasets 2 and 3 over a period of 90 days. Datasets 1, 2 and 3 used initial blood volumes of 500, 1000 and 2000 units respectively. On day 90, the GRASP algorithm depicted a cumulative import amount of 1097 WB units with most of the imports being attributed to blood type O^-. Due to the time frame being restricted to just 90 days, it can therefore be predicted that the level of importation followed an exponential growth, and more importation would occur in the future. Results also indicated that the GRASP algorithm struggled to handle blood types of a higher rarity such as blood type O^-. Overall, five datasets were implemented, but Dataset 2's results could only be acquired.

The study in [7] implemented 7 datasets, however Datasets 6 and 7 could only be examined due to the time period replicating this study's range of 365 days. Dataset 6 used an initial volume of 500 WB units whilst Dataset 7 used 1000 units, both of the datasets used fixed percentage bounds ranging between 25-75%. In [7], the PSO algorithm was implemented to solve the BAP in conjunction with the 7 datasets. These datasets could be compared to Datasets 1 and 6 in the current study as these are similar with regard to parameters. Table 21 illustrates the average results attained from Dataset 1 in this study and Dataset 6 in [7] study

Table 21: Comparison between the results obtained in [7] and the current study per blood type

Study	A^+	A^-	B^+	B^-	AB^+	AB^-	O^+	O^-
The current study (GA result)	0.00	0.00	0.08	0.01	0.28	0.01	0.02	0.00
PSO result [7]	0.00	0.30	0.00	0.05	0.00	0.01	0.41	1.03

Table 21 depicts similar averages attained per blood types expect blood types O^+ and O^-. The study by [7] also presented much larger averages attained for these blood types with a total average of 1.8 WB unit import, whilst the current study only recorded an average of 0.4 WB unit import. The drastic difference can be contributed to the current study utilising the previous days remaining WB units which was not a method implemented in [7].

The current study and research in [7] both examined the metaheuristic algorithms with a larger initial volume of WB units, however [7] used 1000 WB units, whilst the current study used 5000 units. Table 22 below compares the results obtained between the two studies.

Table 22: Comparison between the results obtained in [7] and the current study per blood type for a larger initial WB unit volume

Study	A^+	A^-	B^+	B^-	AB^+	AB^-	O^+	O^-
The current study (SOS result)	0.00	0.00	0.00	0.00	0.62	0.00	0.00	0.00
PSO result [7]	0.00	0.98	0.00	0.17	0.00	0.01	1.22	3.08

The current study recorded that the best metaheuristic implementation for Dataset 5 was the SOS algorithm, as it incurred no importation except for blood type AB^+. The study in [7] displayed a total average of 5.46 WB units. Overall the comparison between these two studies has justified that the act of using the previous day's remaining WB units to treat patients greatly reduces the levels of importation.

The study in [7] was derived from the work done in [3] in 2012. Dataset 1 in [3] used an initial WB units volume of 500 blood units, as well as percentage bounds ranging between 25-75%. Likewise, Dataset 3 used a much larger initial volume of WB units (2000 units) with identical percentage bounds to Dataset 1. Comparisons were therefore made to dataset 5 of this study

Table 23: Comparison between the results obtained in [3] and the current study per blood type with an initial volume of 500 WB units.

Study	Dataset	A^+	A^-	B^+	B^-	AB^+	AB^-	O^+	O^-
The current study (GA result)	1	0.00	0.00	0.08	0.01	0.28	0.01	0.02	0.00
HC result [3]	1	0.00	0.42	0.00	0.11	0.00	0.01	0.67	2.02

Averages obtained from [3] were evaluated over a period of 90 days with blood type O^+ and O^- having the lowest averages with regard to importation levels. The GA implementation in this study out-performed the HC algorithm, however the GA algorithm did experience higher averages for blood types B^+ and AB^+ when compared to the results of [3].

Table 24: Comparison between the results obtained in [3] and the current study per blood type for a larger initial WB unit volume

Study	Dataset	A^+	A^-	B^+	B^-	AB^+	AB^-	O^+	O^-
The current study (SOS result)	5	0.00	0.00	0.00	0.00	0.62	0.00	0.00	0.00
HC result [3]	3	2.73	0.80	1.96	1.03	0.00	0.00	7.14	5.80

In [3], Dataset 3 used an initial WB unit volume of 2000, but still produced much larger averages as compared to the SOS algorithm in this study. Blood types AB^+ and AB^- in [3] recorded no form of importation, whilst the current study incurred imports for blood type AB^+. This could imply that the HC algorithm performs well only for types AB^+ and AB^- when exposed to larger WB unit volumes.

The results examined in this section can be subjected to debate due to the varying parameters for each of the datasets used in relation to the previous literature. Previous work failed to utilise the remaining WB units from the previous day and therefore needed to import additional units to satisfy the demand on a regular basis. The interpretation of the BAP in these studies shows many correlations with respect to data generation, and overall mathematical structure. This study has opted out of the conventional interpretation and has tried to further the research relating to the BAP by covering different mathematical implementations.

Chapter Five

Summary, Conclusion and Future Work

5.1 Summary

The BAP is seen as an optimization problem which tries to efficiently distribute WB units to patients in need, whilst trying to reduce the levels of importation and expiration experienced within the blood bank. Both expiration and importation have expenses associated to these events, and will therefore increase the expenses of a blood bank. Blood was also seen as a precious commodity, therefore expiration of WB units can be seen as the blood bank wasting valuable resources. Due to the BAP containing various components, developing a mathematical model proved difficult, and therefore required certain assumptions to be introduced. The assumptions revolved around physical attributes relating to a WB unit such as lifespan, disregarding human characteristics, ignoring frozen units, etc. In order to verify whether this study produced successful results, an aim and objectives list was generated which incorporated components tailored to the development of the BAP (refer to Section 1.4). The current study successfully completed all the points mentioned in Section 1.4 and ventured into newer territory by implementing different metaheuristic algorithms which differed from previous literature.

The FIFO system for issuing blood was represented by the blood banking policy. There were other sub-components within the issuing system, however the main emphasis was placed on the queueing technique represented in Figure 1. The FIFO system decreased the likelihood of expiration thus allowing the blood bank to utilise WB units efficiently. Five metaheuristic algorithms were used to implement the proposed BAP and these include GA, PSO, DA, SOS and GWO. Prior research made use of the GA and PSO algorithms and subjugated the algorithms in accordance to the BAP. The DA, SOS and GWO are relatively newer techniques with no record of these algorithms being implemented and used to solve the BAP, thus the current study has contributed toward the development of the BAP by means of implementing different metaheuristics as compared to previous work.

Furthermore, most of the literature suffered from the inability of utilising real-world datasets due to confidentiality issues, and therefore generated their own datasets based off scenarios that could be faced by the blood bank. This study incorporated the same data generating techniques, but tried to reduce the randomness by incorporating statistics based on South Africa. Generating percentage bounds in such a manner not only reduces randomness but can also conform to other countries' statistics, for example if a certain country experiences more public holidays in a different month as compared to that of South Africa, then the bounds can be changed accordingly. Finally, the last aspect related to determining the best metaheuristic algorithm which produced the most satisfactory results in relation to the proposed BAP objective function that was discussed in depth in Chapter Four.

5.2 Conclusion

The current study looked at the applications of metaheuristics algorithms coupled with a mathematical model to solve the BAP. Each of the algorithms was subjected to stochastically generated datasets which tested various scenarios that could be imposed upon the blood bank. In addition, this study incorporated South African statistics when generating some of the datasets. This method of dataset generation tries to reduce randomness when generating values, and contributes towards the study of blood management or other related perishable inventory problems. Prior literature merely allocated percentage bounds to each dataset, and did not incorporate any relevant statistics when generating these bounds.

It can be concluded that the SOS algorithm outperformed the other implementations. Whilst the SOS algorithm was by no means the fastest, its results of low importation levels and no form of expiry made up for its lack of speed. The PSO algorithm was deemed as the fastest for producing an output, but suffered with constant low-level imports even after stockpiling occurred. GA and DA did not report significant results, however the GA did perform the best when exposed to Dataset 2, indicating that the GA performs well when exposed to SAGV. The worst algorithm was the GWO due to no form of consistent stock-piling across any of the datasets. The GA and PSO algorithms were implemented in previous literature as reported in [3] and [7] respectively. The study conducted in [3] implemented various hybrid implementations of the GA as well as the HC algorithm, and reported that the HC outperformed the other implementations. With reference to the results obtained by [7], the PSO algorithm seemed to perform well in producing low importation levels, however the algorithm was only run over a course of 90 days and did not incorporate the same model of adding the previous day's remainder to the current day's supply, thus the findings in this study cannot be compared to the research conducted by [7]. Overall the current study has furthered the research relating to the BAP by implementing newer metaheuristics, namely, the DA, SOS and GWO, and incorporated a more consistent and stable method of stochastically generating datasets which can contribute to other relatable inventory management problems. In addition, certain ideas and assumptions from previous literature were expanded upon in this study in order to build a more accurate mathematical model.

5.2.1 Research Contribution

The current study proposed a new mathematical model, that considers the issue of blood compatibility and expiration. More so, three new computational models that are based on the global metaheuristic algorithms namely, DA, SOS and GWO were equally proposed and implemented to solve the BAP. Furthermore, this study attempted to improve upon the aspect of stochastic datasets generation. As mentioned previously, stochastic datasets are used when real world datasets cannot be obtained. The issue with using stochastic datasets relates to the complete randomness when generating values, with the aid of statistics relating to South African public holidays, and schooling terms, the randomness of stochastic dataset was minimized. Even though this technique of dataset generation is not perfected, it is a method open for expansion and further improvement. Lastly, this research also explored the aspect of WB unit expiry which was not considered in many of the previous literatures.

5.3 Future work

It will be interesting to evaluate the performances of the various implementations discussed in this thesis using real-world datasets, especially for locations where accessibility to sensitive data are not hindered. However, if datasets are still being stochastically generated, then possible improvement can be considered from the aspect of utilising more statistical variables when allocating percentage bounds to each month. The mathematical model can also be improved upon, for example, those procedures that requires patients' specific blood types, this accessory could be implemented in future research. Finally, the field of metaheuristics is an ever expanding area, this can also be considered a source of motivation for which even more state-of-the-art metaheuristic algorithms could be tailored for the implementation of various models of the BAP.

References

[1] Hesse S, Coullard C, Daskin M, Hurter A, A case study in platelet inventory management. In Proceedings of the Industrial Engineering Research Conference 6(1), 1997, (pp. 801-806).

[2] Basu. D, Kulkarni. R, Overview of blood components and their preparation, Indian Journal of Anaesthesia 58 (5), 2014, (pp. 529-537).

[3] Adewumi A, Budlender N, Olusanya M, Optimizing the assignment of blood in a blood banking system: some initial results, In Evolutionary Computation (CEC) IEEE Congress, 2012, (pp. 1-6).

[4] Kaveh. A, Ghobadi. M, A multistage algorithm for blood banking supply chain allocation problem, International journal of Civil Engineering 15(1), 2015, (pp. 103-112).

[5] South African Blood Services, https://sanbs.org.za/product-price-list/, [Accessed on: 17-07-2018]

[6] Haijema. R, Wal. J, van Dijk. N, (2005), Blood platelet production: Optimization by dynamic programming and simulation, Computers & Operations Research 34(3), 2007, (pp. 760-779).

[7] Olusanya M. O, Arasomwan MA, Adewumi A. O. Particle swarm optimization algorithm for optimizing assignment of blood in blood banking system. In Computational and Mathematical methods in medicine, 2015, (pp. 1-12).

[8] Adewumi A. O., & Olusanya, M. O, Using Metaheuristic techniques to optimize the Blood Assignment Problem. In Advance Computing Conference (IACC), IEEE International, 2014, (pp. 1331-1336).

[9] Giangrande. P.L.F, The history of blood transfusion, British journal of Haematology 110 (4), 2001, (pp. 758-767).

[10] Reid ME, Lomas-Francis C, Olsson M. L., The blood group antigen factsbook. Academic press; 2012, (pp. 9-12).

[11] Albdulwahab U.S, blood platelet bank inventory management: an approximate dynamic programming approach, 1996, Theses dissertation. (pp. 9-12).

[12] Hardwick. J, Blood storage and transportation, ISBT science series 3(2), 2008, (pp. 177-196).

[13] Chaudhari. C. N, Frozen red blood cells in transfusion, Med J Armed Forces India 65(1), 2009, (pp. 55-58).

[14] Sorensen. K, Metaheuristics- The metaphor exposed, International Transactions in operational research 22(1), 2013, (pp. 3-18).

[15] Radalo. A, Pillay. N, An investigation of dynamic fitness measure for genetic programming 92(1). Expert systems with application, 2017, (pp. 52-72).

[16] Pillay. N, A survey of school timetabling research, Spring Science + business media 218 (1), 2013, (pp.261-293).

[17] Dias. T, Sousa. J, Cunha. J, A Genetic Algorithm for the Bus Driver Scheduling Problem, The Journal of the Operational Research Society 53 (3), 2001, (pp. 324-335).

[18] Bonyadi. M.R, Michalewicz. Z, Particle swarm optimization for single objective continuous space problems: A review, Evolutionary Computation 25(1), 2017, (pp. 1-54).

[19] Liu. Z, Investigation of Particle Swarm Optimization for Job Shop Scheduling Problem, Third International Conference on Natural Computation 5(1), 2007, (pp. 453-458).

[20] Biyanto. T.R, Fibrianto H.Y, Nugroho G, Listijorini E, Budiati T, Huda H, Duelist Algorithm: An algorithm by how duellist improve their capabilities in a duel, Neural and Evolutionary Computing, Advances in Swarm Intelligence 7th International Conference, 2016, (pp. 39-47).

[21] Biyanto. T.R, Fibrianto H.Y, Duelist Algorithm: An Algorithm in Stochastic Optimization Method, Research Centre for Metrology, International Conference on Computer Control, Informatics and Its Applications, 2015, (pp. 1-5).

[22] Ezugwu, A.E., Adeleke, O.J. and Viriri, S, Symbiotic organisms search algorithm for the unrelated parallel machines scheduling with sequence-dependent setup times. PLOS one 13(7), 2018, e0200030.

[23] Cheng M.Y., Prayogo D. and Tran D.H, Optimizing multiple-resources leveling in multiple projects using discrete symbiotic organisms search. Journal of Computing in Civil Engineering, 30(3), 2015, (pp. 1-9).

[24] Venkatakrishnan. G.R, Rengaraj. R, Salivahanan. S, Grey Wolf Optimizer to Real Power Dispatch with Non-Linear Constraints, Department of Electrical and Electronics Engineering, SSN College of Engineering, Chennai, India, CMES 115(1), 2018, (pp 25-45).

[25] Goodnough. L.T, Shander. A, Patient blood management, Anesthesiology 116(1), 2012, (pp. 1367-1376).

[26] Banthao. J, Phongchai. J, An analysis of alternative blood bank locations with emergency referral, Proceedings of the world congress on Engineering and Computer Science 2(1), 2012, (pp. 978-988).

[27] Rusman. M, Rapi. A, Blood banks location model for blood distribution planning in Makassar City, 5th Asia Pacific Industrial Engineering and Management Systems Conference, 2014, (pp. 1323-1237).

[28] Igwe K, Olusanya M, Adewumi A. On the performance of GRASP and dynamic programming for the blood assignment problem. In Global Humanitarian Technology Conference, 2013, (pp. 221-225).

[29] STATS SA- Statistics of South Africa, https://www.statssa.gov.za/publications/P0302/P03022016.pdf [Accessed on 04-06-2018].

[31] World Atlas, https://www.worldatlas.com/articles/countries-with-the-highest-rates-of-hiv-aids.html [Accessed on: 23-07-2018].

[32] News24, https://www.news24.com/SouthAfrica/News/easter-road-death-toll-up-by-14-from-last-year-20180417, [Accessed on 04-06-2018].

[33] School terms South Africa, http://www.schoolterms.co.za/2018.html, [Accessed on 04-06-2018].

[34] Groth. S, Manual on the management, maintenance and use of blood cold chain equipment, World Health Organization: Geneva, 2005, (pp. 5).

[35] Carr. J, An introduction to genetic algorithms, Journal of Computer and Communications 4(7), 2014, (pp. 2-6).

[36] Rini. D.P, Shamsuddin. S.M, Yuhaniz. S.S, Particle swarm optimization: Technique, System and Challenges, International journal of computer applications 14(1), 2011, (pp.19-27).

[37] Prakash. S, Rajathy. R, Implementation of Symbiotic Organism Search algorithm for Extracting Maximum Power from the PV system under Partially Shaded Condition, International science press 8(5), 2015, (pp.1871-1880).

[38] Cheng. M.Y, Prayogo. D, Symbiotic Organism Search: A new metaheuristic optimization algorithm, Computers and Structures 139(1), 2014, (pp.98-112).

[39] Mirjalili S, Mirjalili S.M, Lewis. A, Grey Wolf Optimizer, Advances in Engineering software 69(1), 2014, (pp.46-61).

[40] Shoghian. S, Maryam. K, A Comparison among Wolf Pack Search and Four other Optimization Algorithms, World academy of science engineering and technology 6(1) 2012, (pp. 12-23).

[41] Tetteh. G. A, Optimal allocation of blood products, New Jersey Science and Technology University: PhD Dissertation, 2008, (pp. 1 - 90)

[42] Salehipour. A, Sepehri. M, Exact and heuristic solutions to minimize total waiting time in the blood product distribution problem, Advances in operation research vol. 2012, 2012, (pp. 1 - 25).

[43] Silva. F, Cezarino. W, Salviano. G.R, A decision-making tool for demand forecasting of blood components, IFAC Proceeding Volumes 45(6), 2012, (pp. 1499 - 1504).

[44] Sarkar. S, Artificial blood, Indian journal of critical care medicine 12(3), 2008, (pp. 140 - 145)

[45] Anilkumar. D, Sudashan. P, Ragavan. K, Badu. M.N, A review of artificial blood, International journal of pharmaceutical, chemical and biological sciences 5(2), 2015, (pp. 477 - 480)

[46] Brecher M. E, Shauna. M.D, Hay. M.P.H, ABO Blood type and longevity, American journal of clinical pathology 35(1), 2011, (pp. 96 - 98)

[47] Mansur. A, Vanany. I, Arvitrida. N.I, Challenge and research opportunity in blood supply chain management: A literature review, Conference: 2nd international conference on engineering and technology for sustainable development, 2017, (pp. 1 - 5)

[48] Kim. T. Y, Jeong. M. A, Patient blood management: Anaesthesiologist's perspectives, Hanyang medical review 38(1), 2018, (pp. 49 - 55)

[49] Sundberg. S, Demand forecast process as a part of inventory management, Degree programme in international business, 2009, (pp. 1 - 49)

[50] Haijema. R, Van DerWal. J, Dijk. N, Blood platelet production: Optimization by dynamic programming and simulation, Computer Operation Research 34(3), 2007, (pp. 760 - 779)

[51] Ensafian. H, Yaghoubi. S, Yazdi. M. M, Raising quality and safety of platelet transfusion services in a patient-based integrated supply chain under uncertainty, Computers and Chemical Enguneering vol. 106, 2017, (pp. 355 - 372).

[52] Dillion. M, Fabricio. O, Abbasi. B, A two-stage stochastic programming model for inventory management in the blood supply chain, International journal of production economics vol. 187. 2017, (pp. 27 - 41)

[53] Gear. J. H, The history of blood transfusion, The Leech, 1938, (pp. 61 - 64)

[54] Mittal. N, Singh. U, Sohi. B.S, Modified grey wolf optimizer for global engineering optimization, Applied computational intelligence and soft computing vol. 2016, 2016, (pp. 1 - 16)

[55] Costa. S.D, Sasanti. G.F, Musyafa. A, Soeprijanto. A, Biyanto. T.R, Duellist algorithm for optimisation of safety instrumented system at distillation column based on RAMS+C, Safety and Reliability vol. 2018, 2018, (pp. 1 - 18)

[56] Talukder. S, Mathematical modelling and application of Particle Swarm Optimization, Mathematical modelling and simulation- Msc dissertation, 2011, (pp. 1 - 65)

[57] Choe. S, Li. B, Ri. L, Paek. C, Rim. J, Yun. S, Improved hybrid symbiotic organism search task-scheduling algorithm for cloud computing, KSII transactions on internet and information system 12(8), 2018, (pp. 3516 - 3542)

[58] Xu. H, Liu. X, Su. J, An imporved grey wolf optimizer algorithm integrated with Cuckoo Search, IEEE conference on intelligent data acquisition and advanced computing system, 2017, (pp. 1 - 4)

[59] Beasley. D, Bull. D.R, Martin. R.R, An overview of genetic algorithms part 1: Fundamentals, University computing 15(2), 1993, (pp. 58 - 69)

[60] Najafi. M, Ahmadi. A, Zolfagharinia. H, Blood inventory management in hospitals: Considering supply and demand uncertainty and blood transshipment possibility, Operations research for health care vol. 15, 2017, (pp. 43 - 56).

Appendix A

User Manual

The metaheuristic algorithms were implemented using the Java programming language on the Eclipse Neon IDE version 4.6.0. The program can be run by simply opening the JAR file which will present a GUI as depicted in fig. A.0.1.

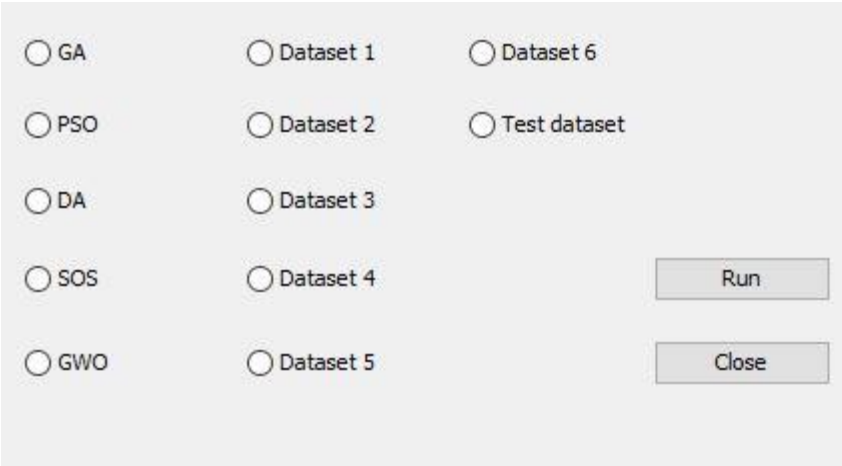

Fig. A.0.1: Representation of the GUI used to run the metaheuristic algorithms

The GUI presents the user with 12 radio buttons split into 5 radio buttons for the metaheuristic algorithms, and 7 radio buttons for the datasets. Also on the GUI are 2 buttons for running and closing the application.

- First click on the radio button for the metaheuristic you wish to run.
- Next click on the dataset you wish to test.
- Finally click the run button to start the algorithm.

Following these steps will open a blank window. The window will only display the final results obtained once the all the days have been completed (365 days). Most of the algorithms took around 60+ minutes to complete therefore it will take a while before all the days results are displayed. Due to the large number of generations per algorithm, it seemed best to only display the best results achieved per day. It is advisable to select the “test dataset” in conjunction with any metaheuristic algorithm as the test dataset only runs over a period of 10 days with fewer iterations etc. which will complete much faster, but offer poorer results. Below is fig. A.0.2 which illustrates the layout of the results once all the days have been completed.

***************************DAY 5***************************

	A+	A-	B+	B-	AB+	AB-	O+	O-
Demand:	74	12	28	5	7	2	90	16
Supply:	94	15	35	6	9	3	115	21
Import:	0	0	0	0	0	0	0	0
Expiry:	0	0	0	0	0	0	0	0

***************************DAY 6***************************

	A+	A-	B+	B-	AB+	AB-	O+	O-
Demand:	80	13	30	5	8	3	98	18
Supply:	86	14	32	5	8	3	105	19
Import:	0	0	0	0	0	0	0	0
Expiry:	0	0	0	0	0	0	0	0

Fig. A.0.2: Illustrates the output representation for a metaheuristic algorithm

Fig A.0.2 depicts 8 columns with each column representing a specific blood type. On the left are 4 variables namely demand, supply, import, and expiry. The values corresponding to both the row and column convey upon the result achieved for that particular day per blood type. The blood types that are more common in society will have relatively higher values, and the supply per blood type should increment as time progresses due to the stock-piling effect. Finally the algorithm terminates after 365 days have completed and displays the averages attained per variable.