– Proceedings –

41st Annual Conference of the Operations Research Society of South Africa

16–19 September 2012
Aloe Ridge Hotel, Muldersdrift, South Africa

ISBN: 978-07972-1406-4

Proceedings of the 41st Annual Conference of the Operations Research Society of South Africa

ISBN: 978-07972-1406-4

Editorial Board

Editor-in-chief:

Associate Editors:

Review process

Nineteen (19) manuscripts were submitted for possible inclusion in the *Proceedings of the 41th Annual Conference of the Operations Research Society of South Africa, 2012*. All submitted papers were double-blind peer-reviewed by at least two independent reviewers. Papers were reviewed according follow criteria: technical quality, i.e. correct use of language, clarity of expression, quality and justification of arguments; and on the contribution to Operations Research, i.e. knowledge of field, quality and consistency of referencing, significance of contribution and suitability for conference proceedings. Of the nineteen (19) submitted papers, fourteen (14) were ultimately, after consideration and incorporation of reviewer comments, judged to be suitable for inclusion in the proceedings of the conference. The proceedings is published online at
`http://www.orssa.org.za/wiki/uploads/Conf/2012ORSSAConferenceProceedings.pdf`.

Reviewers

The editorial would like to thank the following reviewers:

Wilna Bean	Council for Scientific and Industrial Research, South Africa
Ian Campbell	University of the Witwatersrand, Johannesburg, South Africa
Wenwei Cao	Georgia Tech, USA
Louis Danhauser	Vodacom, South Africa
Dave Evans	Development bank of Southern Africa
Hans Ittmann	HWI Consulting, South Africa
Pieter Fourie	Swiss Federal Institute of Technology Zurich, Switzerland
Prof. Johan Joubert	University of Pretoria, South Africa
Renee Koen	Council for Scientific and Industrial Research, South Africa
Brian Kues	Georgia Tech, USA
Dr. David Lubinsky	OPSI Systems, South Africa
Dr. Elias Munapo	UNISA, South Africa
Winnie Pelser	Armscor, South Africa
Angela Rademeyer	OPSI Systems, South Africa
Dr. Peter Scmithz	Council for Scientific and Industrial Research, South Africa
Dr. Esbeth van Dyk	Council for Scientific and Industrial Research, South Africa
Nadia Viljoen	Council for Scientific and Industrial Research, South Africa
Elias Willemse	LTS Consulting, South Africa

Elias J Willemse
(e) ejwillemse@gmail.com
(t) +27 71 890 2714
Editor-in-Chief: ORSSA 2012 Proceedings

Table of contents

Evert B Schlünz, Pavel M Bokov & Rian H Prinsloo, *Application of a harmony search algorithm to the core fuel reload optimisation problem for the SAFARI-1 nuclear research reactor* 1

Sandile Saul & Aderemi O Adewumi, *An artificial bees colony algorithm for the Travelling Tournament Problem* 10

Benson Y Baha, Aderemi O Adewumi, Nachamanda V Blamah & Gregory M Wajiga, *An artificial neural network for detecting risk of Type 2 Diabetes* 20

Jacques du Toit & Jan H van Vuuren, *Coastal threat evaluation decision support* 31

Kanshukan Rajaratnam & Chun-Sung Huang, *Consumer Loan Decisions with Profit-Loss Tradeoff under Multiple Economic Conditions* 40

Moonyoung Yoon & Tanya Lane-Visser, *Finding the best pass-receiving position in the RoboCup Small-Size League* 48

Nadia M Viljoen, *An implementable routing solution for home-based care in South Africa* 58

Frans J Snyders & Tanya Lane-Visser, *Improving the work rate of community health workers through optimisation* 67

Nadia M Viljoen, Julie Swann, Melih Celik, Wenwei Cao & Ozlem Ergun, *Keeping it simple in a data-sparse environment: The case of donor breastmilk demand and supply in South Africa* 78

Sivashan Chetty & Aderemi O Adewumi, *Results of local search heuristics for the Annual Crop Planning Problem* 83

Mark D Einhorn, Jan H van Vuuren & Alewyn P Burger, *Self-organising traffic light control* 93

Emmanuel Dufourq, Michael O Olusanya & Aderemi O Adewumi, *Studies in metaheuristics for the Blood Assignment Problem* 107

Nelishia Pillay & Christopher Rae, *A survey of hyper-heuristics for the Nurse Rostering Problem* 117

Johanna H Nel & Stephan C Krygsman, *The use of Global Positioning Systems in travel surveys: Experience from a pilot project* 125

Proceedings of the 2012 ORSSA Annual Conference
pp. 1–9
www.orssa.org.za/wiki/uploads/Conf/2012ORSSAConferenceProceedings.pdf

ORSSA Proceedings
ISBN
978-07972-1406-4
©2012

Application of a harmony search algorithm to the core fuel reload optimisation problem for the SAFARI-1 nuclear research reactor

EB Schlünz[*] PM Bokov[†] RH Prinsloo[‡]

Abstract

The *core fuel reload optimisation problem* (CFROP) refers to the problem of finding an optimal fuel loading configuration for a nuclear reactor core. The CFROP is typically multiobjective, nonlinear, discrete and combinatorial in nature. In this paper, a mathematical formulation of the CFROP for the SAFARI-1 nuclear research reactor is presented. The multiple objectives applicable to SAFARI-1 are aggregated into a single objective function. A harmony search algorithm with a dimensionality reduction procedure is proposed as a solution technique for solving the problem approximately. The algorithm has been implemented on a personal computer and applied to solve the CFROP for a historic SAFARI-1 core. Fuel loading configurations that improve upon the historically chosen configuration were obtained by the algorithm. The results show that the solution approach has the capability of proposing good fuel loading configurations to assist the operators of SAFARI-1.

Key words: Combinatorial optimisation, metaheuristics, core fuel reloading, harmony search

1 Introduction

One of the tasks that occurs between operational cycles of a nuclear reactor, is the reloading of *fuel assemblies* (FAs) in the core. Typically, depleted FAs are replaced by fresh FAs and the placement of all FAs may be changed, resulting in a core reconfiguration. The *core fuel reload optimisation problem* (CFROP) refers to the problem of finding an optimal fuel loading configuration for a nuclear reactor core. The characteristics of the CFROP comprise high dimensionality, discrete variables, nonlinear and nonconvex functions, computationally expensive function evaluations and disconnected feasible regions in the search space [8]. These characteristics, as well as the multiobjective and combinatorial nature of the problem, clearly demonstrate that the CFROP is a difficult, ill-structured problem to solve.

[*] Corresponding author: Radiation and Reactor Theory, Necsa, PO Box 582, Pretoria, 0001, fax: 012 305 5166, email: bernard.schlunz@necsa.co.za

[†] Radiation and Reactor Theory, Necsa, email: pavel.bokov@necsa.co.za

[‡] Radiation and Reactor Theory, Necsa, email: rian.prinsloo@necsa.co.za

A number of solution techniques have been proposed in order to solve the problem, such as mathematical programming methods, expert systems, simulated annealing, evolutionary algorithms, swarm intelligence algorithms and tabu search [6]. However, the overwhelming majority of CFROP research has been orientated towards power reactors. Research reactors pose different challenges than power reactors, *e.g.* different utilisation requirements affect the type and number of objective functions, and different core designs affect the core symmetry and fuel depletion distributions.

Some of the research reactor CFROP studies are briefly presented here. Mahlers [4] developed an algorithm based on successive mixed-integer linear programming. A single objective CFROP model with a linearity approximation was used, where the goal was to maximise the thermal neutron flux (the number of neutrons in the lower energy spectrum that flows through a unit area in unit time) in experimental facilities in the core. Van Geemert *et al.* [9] applied multiple cyclic interchange algorithms to their single objective, safety constrained, CFROP model in order to solve the problem. Mazrou & Hamadouche [5] used an *artificial neural network* (ANN) to predict two safety parameters (aggregated into a single objective function value) and simulated annealing to optimise the loading configuration. Finally, Hedayat *et al.* [3] also employed an ANN in order to predict operational and safety parameters. They used a state-of-the-art multiobjective genetic algorithm, called NSGA-II, for solving the CFROP.

In this paper, a metaheuristic algorithm called *harmony search* (HS) [2] is proposed as a solution technique for solving the CFROP for the SAFARI-1 nuclear research reactor. The paper is organised as follows. Section 2 contains a description of the CFROP for SAFARI-1 and its mathematical problem formulation. In Section 3, the classical HS algorithm is described, along with the algorithmic implementation of an adapted HS algorithm that is proposed in this work. The results that were obtained from the application of the proposed HS algorithm to solve the CFROP for SAFARI-1 are presented in Section 4, followed by a conclusion and thoughts for future research in Section 5.

2 The problem description

SAFARI-1 is primarily utilised for scientific research, radiopharmaceutical isotope production, irradiation services and materials testing. The core layout of the SAFARI-1 model that was used in this work, is presented in Figure 1. The proposed HS algorithm attempts to optimise the placement of 26 FAs in the 26 red coloured positions, shown in Figure 1. There are at most $26! \approx 4\times10^{26}$ different FA loading configurations. Each FA has a unique fuel depletion, characterised by the assembly's Uranium-235 (^{235}U) mass. The contribution from each FA to the neutron flux in the core is dependent on the fuel depletion, and current and historical positions of the FA.

Three utilisation goals are pursued during a typical operational cycle of SAFARI-1, while a number of safety constraints need to be adhered to. The first goal is to maximise the production of the radioisotope Molybdenum-99 (^{99}Mo), which is used for medical diagnostic purposes. The second goal is to maximise the silicon doping (the intentional introduction of impurities into pure silicon) capacity of the reactor for the production of silicon semiconductors used in electronic equipment. Finally, the third goal is to maximise the neutron intensity in the neutron beams which are placed next to the core and used for various research purposes.

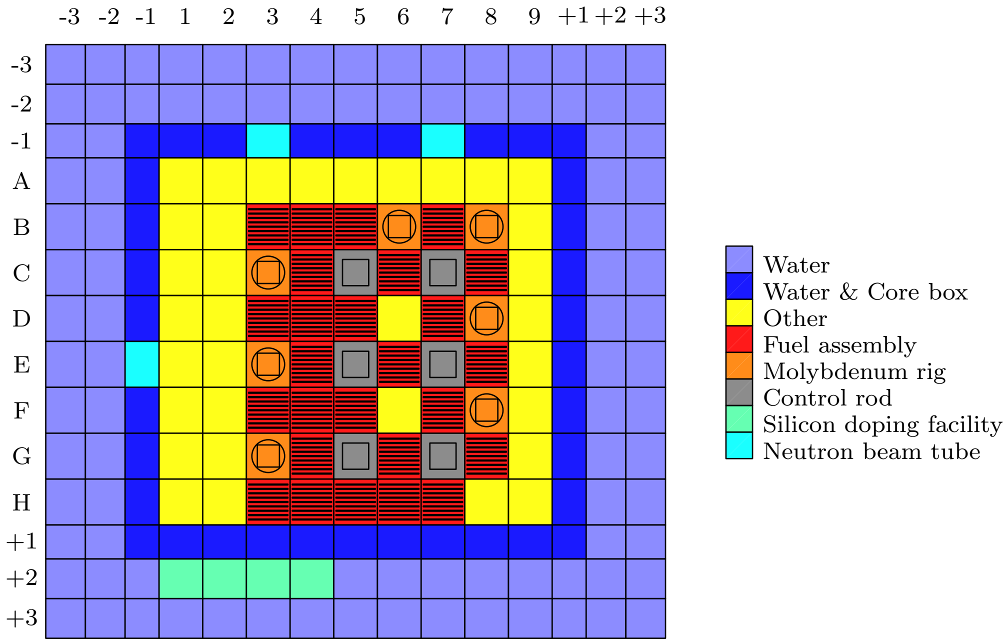


Figure 1: The SAFARI-1 core layout.

When these goals are translated into physical requirements, the first goal corresponds to the maximisation of the power levels in the ^{99}Mo rig positions in the reactor (indicated in orange in Figure 1). The second and third goals correspond to the maximisation of the thermal neutron flux in the silicon doping facility (indicated in green in Figure 1) and in the entry points to the neutron beams, also known as beam tubes (indicated in light blue in Figure 1).

In order to evaluate the objectives and constraints for a FA loading configuration, a reactor core calculational system is required. Essentially, such a system calculates the neutron distribution and material evolution in the core during nuclear reactor operation. In this study, the OSCAR-4 [7] system was utilised. A core simulator within OSCAR-4 solves the three-dimensional, multi-energy group, time-independent diffusion equation [1] with time-dependent parameters to determine the neutron flux distribution throughout the reactor core during an operational cycle. This calculated flux distribution is used in secondary physical models to determine operational and safety related parameters in the core (which may appear as CFROP objectives or constraints). Only numerical values for these parameters are returned, resulting in so-called "black-box" function/constraint evaluations for a CFROP.

2.1 Mathematical problem formulation

Let n denote the number of FAs, and the number of fuel loading positions in the reactor core. Furthermore, let $\mathcal{I} = \{1, \dots, n\}$ denote the set of FA indices, and let $\mathcal{J} = \{1, \dots, n\}$ denote the set of fuel loading position indices. A loading configuration is then defined as a permutation vector $x = (x_1, \dots, x_n)$ of n decision variables, where $x_j \in \mathcal{I}$ corresponds to an integer value representing the index of the FA placed in position $j \in \mathcal{J}$ in the core.

Let $f_M(x)$ denote the percentage of total reactor power in the ^{99}Mo rig positions, let $f_S(x)$

denote the thermal neutron flux in the silicon doping facility, and let $f_1(x), f_2(x)$ and $f_3(x)$ denote the thermal neutron flux in the three beam tubes. In order to optimise these quantities simultaneously, the *scaled linear weighted sum* aggregation approach is followed whereby all five objectives are combined into a single objective function. This approach is an initial attempt to solve the CFROP for SAFARI-1 and the results will produce a baseline for future research.

Let $\mathcal{K} = \{M, S, 1, 2, 3\}$ denote the set of objective indices. Each objective $k \in \mathcal{K}$ is associated with a relative importance weight w_k, as well as a scaling factor p_k. The weights should reflect the decision maker's preferences regarding the importance of each objective, relative to the other objectives. The scaling factors are necessary in order to create dimensionless function values that are equally scaled by order of magnitude. Ideally, these scaling factors should be chosen as $p_k = f_k^*$, where f_k^* is the optimal objective value to the single objective problem for objective $k \in \mathcal{K}$. However, since these optima are rarely known, they may be replaced by some target values, or at least by some values within the same range as each objective value. Therefore, expert knowledge has been employed in this study in order to determine the weights w_k and scaling factors p_k for the objectives. These values will be reported in Section 4. According to the chosen aggregation approach, the objective of the CFROP for SAFARI-1 is to maximise

$$F_0(x) = \sum_{k \in \mathcal{K}} w_k \left(\frac{f_k(x)}{p_k} \right). \tag{1}$$

The CFROP for SAFARI-1 is subject to three safety constraints. The first constraint is to maintain the relative *power peaking factor* (PPF) of the core below a certain threshold. The PPF is a measure of the maximum relative power obtained anywhere in the fuel. If $P(x)$ denotes the PPF and $P_{\max}$ denotes the threshold value, then the power peaking constraint is $P(x) \leq P_{\max}$. Two safety parameters, termed the *control bank worth* (CBW) and the *shutdown margin* (SM), are utilised to ensure that the efficiency of the control system (in this case rods) is adequate. The CBW and SM refer to the total and operationally available neutron absorption capability of the control rods, respectively, and should be maintained above defined thresholds. The bank worth constraint is given by $B(x) \geq B_{\min}$, where $B(x)$ denotes the CBW and $B_{\min}$ the threshold value, and the shutdown margin constraint by $S(x) \geq S_{\min}$, with $S(x)$ and $S_{\min}$ denoting the SM and threshold value, respectively.

The safety constraints are defined as soft constraints in order to have an unrestricted search space for the solution technique. If a candidate solution violates any of the safety constraints, a corresponding penalty value is incurred for that violation, and added to the objective function value. This penalty value is related to the magnitude of the violation and is always greater than 1. If the power peaking constraint is violated, its corresponding penalty value, $\hat{P}$, is calculated as $\hat{P} = P(x)/P_{\max}$, or zero otherwise. If the bank worth constraint is violated, its corresponding penalty value, $\hat{B}$, is calculated as $\hat{B} = B_{\min}/B(x)$, or zero otherwise. Finally, if the shutdown margin constraint is violated, its corresponding penalty value, $\hat{S}$, is calculated as $\hat{S} = S_{\min}/S(x)$, or zero otherwise. Accordingly, objective function (1) of the CFROP for SAFARI-1 is extended to maximise

$$F(x) = \sum_{k \in \mathcal{K}} w_k \left(\frac{f_k(x)}{p_k} \right) - \hat{P} - \hat{B} - \hat{S}. \tag{2}$$

According to the above definitions of the penalty values, dimensionless penalty values are created that are equally scaled by order of magnitude, similar to the scaled values of the individual objectives. The penalty values have equal weights since it is equally important for every constraint to be satisfied in order to obtain a feasible final solution. The requirement of having a feasible final solution originates from a nuclear regulatory point of view — a reactor may only be operated within its safety limits.

3 The harmony search solution approach

A metaheuristic algorithm called harmony search [2] is proposed as a solution technique for solving the CFROP for SAFARI-1. A metaheuristic solution approach was chosen because of the ill-structured nature of the problem, as described in Section 1, and the necessity of using black-box function/constraint evaluations, as described in Section 2.

The HS algorithm is inspired by the musical phenomenon of *harmony* — an aesthetically pleasing combination of sounds. Its simple structure and capability of solving discrete and continuous optimisation problems (without the need for major changes) makes it very appealing, since the specific CFROP for SAFARI-1 problem formulation may change over the operational time of the reactor. As a global search algorithm, it may also explore the search space more effectively than local search algorithms would (*e.g.* simulated annealing and tabu search) within the limited computational budgets of CFROPs.

The first step in the HS algorithm is to initialise a memory structure, called the *harmony memory* (HM), with random[1] harmonies (solutions). The second step is to improvise (create) a new solution, variable by variable. For each variable, a value may be randomly selected either from the HM, or from its allowable range. If a variable takes a value from the HM, a secondary process called pitch adjustment may be performed that potentially alters the solution slightly. In the third step of the algorithm, a new solution's objective function value is compared to the solution in the HM with the worst objective function value. If the new solution is found to be better, it replaces the worst solution in the HM. Finally, the algorithm terminates if some pre-defined termination criteria are met; otherwise it returns to the second step.

A parameter called the *harmony memory consideration rate* (HMCR), is introduced in order to bias the choice of variable value z towards values from the HM. Therefore, with probability HMCR, z is randomly selected from the values in the HM corresponding to that variable. Alternately, with probability 1−HMCR, z is randomly selected from its allowable range. Typically, HMCR $\in [0.7, 0.95]$. A parameter called the *pitch adjustment rate* (PAR), determines the probability of performing the pitch adjustment, while the *bandwidth* (BW) determines how much z may be adjusted. Typically, PAR $\in [0.1, 0.5]$. Thus, with probability PAR, a value may be adjusted according to $z_{\text{new}} = z_{\text{old}} + \text{BW}\cdot\varepsilon$, where ε is a random number in the range $[-1, 1]$ and $z \in \mathbb{R}$. For $z \in \mathbb{Z}$, $\varepsilon \in \{-1, 0, 1\}$. Alternately, if z has a discreet set of neighbouring values, the pitch adjustment process will randomly select a neighbouring value, where the number of neighbours are determined by BW. The pitch adjustment may improve a solution or may, together with the non-HM random selection, allow the HS algorithm to escape local optima.

[1] Where any reference is made to a random selection in this paper, the selection is assumed to be performed according to a uniform distribution.

3.1 Algorithmic implementation of the proposed algorithm

The HM, with a chosen size of 15, is initially filled with randomly generated solution vectors — each corresponding to a random permutation of the values $\{1, \ldots, n\}$. The procedure that creates a new solution was adapted for the permutation vector representation of a CFROP solution in order to create unbiased and valid solutions without the need of a repair procedure. Each new position in a partial solution is randomly selected. Then, with probability HMCR = 0.95, an FA for the selected position is chosen from the HM from an allowable set. This set contains all the FAs in the HM corresponding to the position, excluding the FAs already selected in the partial solution. However, if this allowable set is empty, a FA is randomly chosen from the available range. Similar procedures for creating the allowable sets during the pitch adjustment (with PAR = 0.25 and BW = 1) and the non-HM random selection are adopted.

Since OSCAR-4 requires approximately 4 minutes computational time to evaluate a candidate solution, an archive containing all the previous solutions is introduced into the algorithm. Before a solution is passed to OSCAR-4, the archive is searched in order to determine whether the solution has been evaluated before. If so, the function and constraint values are simply retrieved from the archive. The algorithm then terminates when a pre-specified maximum number of iterations (chosen as 1 000) has been performed. Hereafter, this algorithm is referred to as *adapted harmony search* (HS-A).

An attempt was made to reduce the dimensionality of the CFROP as the algorithm progresses by assuming that FAs may be fixed to positions during the search process, thereby reducing the number of decision variables. If a sufficient number of solutions in the HM (more than 70%) contain the same FA in the same position at the end of an iteration, that FA is considered as fixed and that position as "frozen." In all subsequent iterations, the frozen positions retain their fixed FAs and the corresponding indices are removed from the available sets $\mathcal{J}$ and $\mathcal{I}$, thereby reducing the problem dimensionality. HMCR is initially set to 0.75, and increased to 0.85 and 0.95 after 1/3 and 2/3, respectively, of the total number of iterations have been performed. This allows the algorithm to search more globally during the initial phases of the search in an attempt to avoid premature convergence. The algorithm terminates when all loading positions are frozen, or when a pre-specified maximum number of iterations have been performed. Hereafter, this modified algorithm is referred to as *harmony search with dimensionality reduction* (HS-DR).

4 Computational results

The HS-A and HS-DR algorithms, as well as a pure *random search* (RS), were applied to a problem instance of the CFROP for SAFARI-1 based on a historical operational cycle of the reactor. It was suggested by a panel of experts that, for the sake of this study, the five objectives are to be deemed equally important and were therefore assigned the weights $w_k = 1$ for $k \in \mathcal{K}$. Due to confidentiality reasons, the target values (obtained through expert knowledge) assigned to the scaling factors p_k for each of the objectives, are withheld. Each algorithm was used to solve the problem instance 6 times, since HS and RS are stochastic techniques. The solutions obtained from using the algorithms were compared to that of the actual loading configuration, hereafter referred to as the *reference* solution. This reference solution was obtained from historic operational loading data of SAFARI-1, which is a result of an operating strategy that attempts to level the neutron flux profile over the core.

The reference solution for the problem instance, denoted by x_{ref}, attains an objective function value of 5.987. The best solution obtained by HS-A (denoted by x^*_{ad}) attained an objective function value of 6.468 with zero penalty, while HS-DR obtained a best solution (denoted by x^*_{dr}) attaining an objective function value of 6.485 with zero penalty. The RS yielded a best solution (denoted by x^*_{rs}) attaining an objective function value of 6.329 with zero penalty. These solutions correspond to an 8.03%, 8.32% and 5.71% improvement in objective function value, respectively, to that of the reference solution. The algorithms required an average of 2.67 days of computational time (on a personal computer with a 2.66 GHz Intel® Core™ 2 Quad processor, 4.0 GB RAM and a 32-bit operating system) for the 1 000 iterations. HS-DR did not freeze all the positions in any of its 6 executions.

In Figure 2, the convergence graphs of the average incumbent objective function values over the 6 solution instances of HS-A, HS-DR, and the RS are presented. Clearly, both HS algorithms perform significantly better than the RS. HS-A achieved a better convergence rate than HS-DR. This behaviour may be explained by the small HMCR value in the initial phases of HS-DR. As a result, HS-DR did not fully exploit the HM. However, HS-DR obtained a significantly larger number of solution improvements during the final stage of the algorithm's execution than HS-A did (31 versus 6 in the last 300 iterations), and ultimately obtained the better solution (maximum and average) of the two algorithms. The larger number of improvements may be explained, firstly by the fact that HS-DR only fully exploits the HM during the final stage of its execution, and secondly by the dimensionality reduction procedure. Since positions become fixed towards the end of HS-DR's execution (on average, the first position was frozen at iteration 433), it employs a narrower local search, thereby obtaining local optima which were missed by HS-A. Due to the time-consuming nature of this work, detailed parameter studies have not yet been performed to further investigate the HMCR and dimensionality reduction relationship beyond what is reported in this paper.

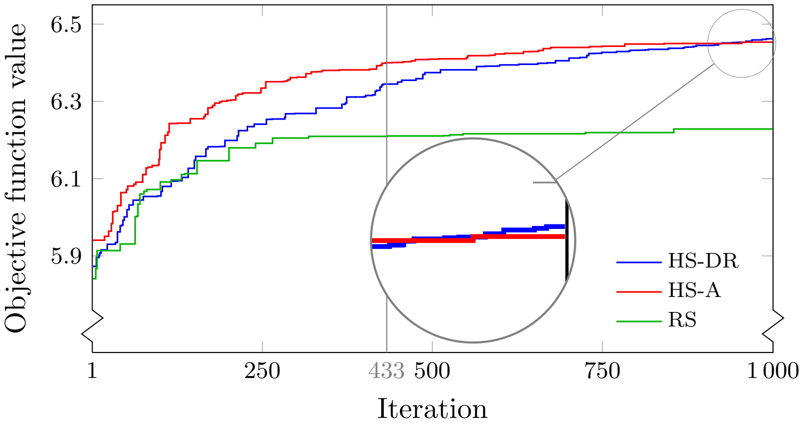


Figure 2: Convergence graphs of the average incumbent objective function values.

The difference between the values of the individual objectives corresponding to the reference solution and x^*_{ad}, as well as the reference solution and x^*_{dr}, are presented in Table 1.

For the sake of completeness, x^*_{rs} is included in Table 1. Both x^*_{ad} and x^*_{dr} attained a slight improvement in the ^{99}Mo objective (f_M), noticeable deterioration in the silicon objective (f_S), insignificant deterioration in the first beam tube objective (f_1), and significant improvements in the second (f_2) and third (f_3) beam tube objectives, when compared to the reference solution. Note that x^*_{ad}, x^*_{dr} and x^*_{rs} were trade-off solutions to x_{ref}.

Objective	x^*_{ad}	x^*_{dr}	x^*_{rs}
f_M	+0.30%	+1.18%	0.00%
f_S	−7.34%	−8.25%	+5.28%
f_1	−0.05%	−0.43%	−3.50%
f_2	+28.12%	+29.82%	+11.20%
f_3	+22.32%	+23.19%	+18.74%

Table 1: Difference between individual objectives corresponding to x_{ref}, and x^*_{ad}, x^*_{dr} and x^*_{rs}.

In Figure 3, the physical core configurations of x_{ref} and x^*_{dr} (the best solution found during this work) are presented visually in terms of the ^{235}U mass (in grams) in each FA. The difference in ^{235}U distribution over the core between the two configurations is quite significant. In x^*_{dr}, the FAs with a high ^{235}U mass are now placed in the positions that are closest to the three beam tubes, thereby causing the significant improvement in the objective values of f_2 and f_3, while deteriorating the objective value of f_S. Although the power levels dropped in the lower ^{99}Mo rig positions due to less ^{235}U mass surrounding it, the objective value of f_M was maintained (even increased slighty) as a result of more ^{235}U mass surrounding the upper ^{99}Mo positions, causing an increase in their power levels.

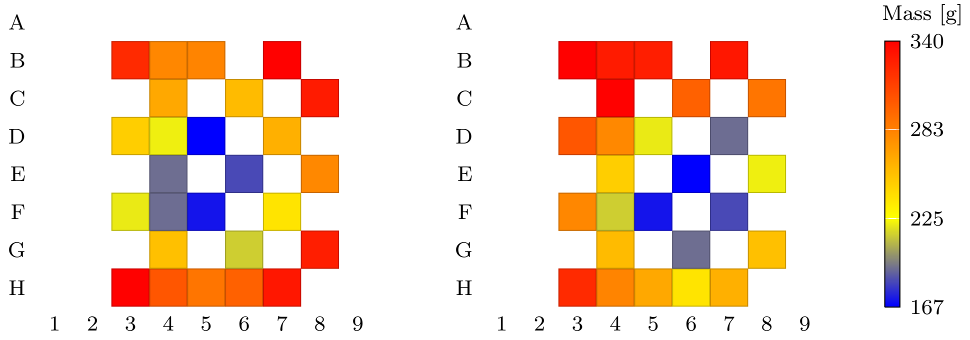


Figure 3: The physical core configurations of x_{ref} (left) and x^*_{dr} (right).

5 Conclusion and future research

In this paper, a mathematical formulation of the CFROP for the SAFARI-1 research reactor was presented, incorporating the specific utilisation objectives of the reactor, along with a number of standard safety constraints. The multiple objectives of the problem were aggregated into a single objective function using the scaled linear weighted sum approach. A modified HS algorithm was developed and, along with a RS, successfully applied to solve the

CFROP for SAFARI-1 approximately. With a computational budget of 1 000 iterations, it was found that HS-DR yielded slightly better solutions than HS-A, with both yielding significantly superior solutions than the RS. The best solution obtained in this work was a trade-off to the reference solution, resulting in an increase of 1.18%, 29.82% and 23.19% in three of the objectives and a decrease of 8.25% and 0.43% in the other two objectives. Therefore, the proposed solution approach may be used effectively to produce good fuel loading configurations for SAFARI-1 from cycle to cycle. Further studies will include an investigation into the dimensionality reduction procedure's effectiveness within smaller computational budgets, the introduction of Pareto-optimality to model the multiobjective nature of the CFROP problem, and approximation strategies to decrease the computational time of function evaluations.

Bibliography

[1] Duderstadt JJ & Hamilton LJ, 1976, *Nuclear reactor analysis*, John Wiley & Sons, New York (NY).

[2] Geem ZW, Kim JH & Loganathan GV, 2001, *A new heuristic optimization algorithm: Harmony search*, Simulation, **76(2)**, pp. 60–68.

[3] Hedayat A, Davilu H, Barfrosh AA & Sepanloo K, 2009, *Optimization of the core configuration design using a hybrid artificial intelligence algorithm for research reactors*, Nuclear Engineering and Design, **239**, pp. 2786–2799.

[4] Mahlers YP, 1997, *Core loading pattern optimization for research reactors*, Annals of Nuclear Energy, **24(7)**, pp. 509–514.

[5] Mazrou H & Hamadouche M, 2006, *Development of a supporting tool for optimal fuel management in research reactors using artificial neural networks*, Nuclear Engineering and Design, **236**, pp. 255–266.

[6] Meneses A, De Lima A & Schirru, 2010, *Artificial intelligence methods applied to the in-core fuel management optimization*, pp. 63–78 in Tsvetkon P (Ed.), *Nuclear Power*, InTech, [Online], [Cited June 6th, 2012], Available from `http://www.intechopen.com/books/nuclear-power`

[7] Stander G, Prinsloo RH, Müller E & Tomašević DI, 2008, *OSCAR-4 code system application to the SAFARI-1 reactor*, Proceedings of the International Conference on the Physics of Reactors 2008 (PHYSOR 08), Interlaken, pp. 1179–1187.

[8] Stevens JG, Smith KS & Rempe KR, 1995, *Optimization of pressurized water reactor shuffling by simulated annealing with heuristics*, Nuclear Science and Engineering, **121**, pp. 67–88.

[9] Van Geemert R, Quist AJ, Hoogenboom JE & Gibcus HPM, 1998, *Research reactor in-core fuel management optimization by application of multiple cyclic interchange algorithms*, Nuclear Engineering and Design, **186**, pp. 369–377.

Proceedings of the 2012 ORSSA Annual Conference
pp. 10–19
www.orssa.org.za/wiki/uploads/Conf/2012ORSSAConferenceProceedings.pdf

ORSSA Proceedings
ISBN
978-07972-1406-4
©2012

An Artificial Bees Colony algorithm for the Traveling Tournament Problem

S Saul[*] Aderemi Adewumi[†]

Abstract

Scheduling of professional sports is one of many researched practical problems in combinatorial optimization. The scheduling of professional sports is a known NP-Hard problem which is very difficult to solve as it involves multiple constraints. This paper addresses the Traveling Tournament Problem(TTP). The goal of TTP is to create feasible sport schedules that minimizes the distance traveled by teams. An Artificial Bee Colony(ABC) algorithm was designed for the problem and the algorithm was applied to different instances. Results obtained are compared with previous results in literature.

Key words: Sport scheduling, Traveling tournament problem, double round robin tournament, Artificial bee colony

1 Introduction

Scheduling is one of many researched practical problems in combinatorial optimization. Scheduling consists of developing a timetable of events within a specified time frame and given a set of resources such that no events conflict with one another or violate any constraints [5]. Some real world scheduling problems include job scheduling, school examination timetable and scheduling processes in a computer operating system.

For many years the design of professional sports schedules has been done by hand through a combination of adherence to tradition and years of experience [13]. However there have been several efforts to understand and examine the goals and constraints of sports scheduling; most of these constraints have focused on the difficulty of producing a feasible schedule. The organization of sports has become more complex now, professional sports leagues now consist of many teams playing long schedules, and thus the creation of sports league schedules has become a difficult task. Beyond the complexity of creating a schedule to be contained within a

[*] School of Mathematics, Statistics & Computer Science, University of Kwazulu Natal, Durban, South Africa, email: sandilesaul704@gmail.com

[†] Corresponding author: School of Mathematics, Statistics & Computer Science, University of Kwazulu Natal, Durban, South Africa, email: adewumia@ukzn.ac.za

specified time period, there are also the issues of venue availability, travel time and cost, fairness in schedules between the teams, along with a multitude of other constraints and desired goals [12]. Devising a schedule that meets all of a league administrator's criteria regarding when and where games may be played is an NP-hard optimization problem, that is, there is no known polynomial time algorithm capable of finding a solution that combines the given resources in an optimal way without violating constraints, because as more teams are added to a league, the search space of possible schedules grows exponentially. Sports scheduling has attracted a lot of interest from different research communities such as operations research and artificial intelligence.

There are various relevant aspects to be considered in determining the best schedule, in some situations one seeks for a schedule that minimizes the total distance traveled (traveling tournament problem) and in some situations one seeks for a schedule that attempts to minimize the number of breaks [6]. The former will be the focus of our research.

2 Problem Description and Formulation

A double round robin tournament is a tournament in which every team plays every other team twice, once at home and once away. The tournament is made up of $2(n-1)$ rounds, n is the number of teams. Each team begins at its home site and travels to play its games at the chosen venues and returns to its home base at the end of the schedule [11]. The input to TTP is an $n * n$ distance matrix D whose d_{ij} denotes the distance between teams T_i and T_j. The objective is to find a schedule with a minimum total distance traveled by all the teams.

$$Minimize \sum_{i=1}^{n} \sum_{k=1}^{2(n-1)} d_{ij} x_{ij}$$

$$x_{ij} = \left\{ \begin{array}{ll} 0, & \text{if home game at round } r_k \\ 1, & \text{if away game at round } r_k \end{array} \right\}$$

The schedule must satisfy the following two constraints:

- **Atmost Constraint**: A team cannot play more than three consecutive home or away games.
- **Norepeat Constraint**: A game between team T_i and T_j at T_i's home cannot be followed by a game between T_j and T_i at T_j's home.

3 Prior and Related Work

There has been substantial work in recent years in the field of sport scheduling, in this section we review some of the work. Hung *et al.* [7] used a genetic algorithm to solve the constraint satisfaction problem of sports schedules and analyzed the battle combination constraints of the National Basketball Association (NBA) and National Hockey League (NHL). They simulated the NBA schedules in 2009. In [7] they planned the schedules based on the shortest traveling cost. The system was able to arrange two or more sports schedules. The schedules satisfied all characteristics of every league and the genetic algorithm used, solved the scheduling problems effectively. Anagnostopoulos *et al.* [1] proposed a simulated algorithm for TTP that explores both the feasible and infeasible schedules. In [1] they used a large neighborhood and advanced techniques such as strategic oscillation and reheats to balance the exploration of the feasible and infeasible regions and to escape local minima. The results they obtained matched the best known results on smaller instances and were much better on larger instances.

Furthermore Wei *et al.* [11] tackled the mirrored version of TTP known as Mirrored Traveling Tournament Problem (mTTP). They proposed a hybrid local search approach based on the combination of tabu search and a variable neighborhood descent metaheuristic together with a greedy randomized adaptive search procedure which explores a large neighborhood with effective moves. mTTP is a generalization of TTP, the main difference is the concept 'mirrored double round robin', which is basically a tournament where each team play every other team once in the n-1 rounds, followed by the same number of games with reversed venues in the last n-1 rounds [11]. Their proposed approach did not perform well on rather small instances, due to the briefness of neighborhood structures. Gaspero and Schaerf [4] proposed a tabu search approach to the solution of TTP that makes use of a combination of two neighborhood relations. In [4], before applying a local search to the problem, they had to define several features. These features include illustrating the search space, the cost function and the procedure for computing the initial state. Then finally, they defined the neighborhood structure and described the guiding metaheuristic. Their algorithm proved to be quite robust leading to a very compact distribution of solution cost.

4 Methodology

ABC algorithm was investigated and presented in this paper. First we described how initial schedules are created, followed by the neighborhoods explored and then the optimization technique(ABC).

4.1 Initialization

We used a polygon method as in [3] to build a schedule for a single round robin tournament with n abstract teams, without assigning the venues. The abstract teams are randomized rather than a simple increasing order from 1 to n. The abstract teams ordered 1 to $n-1$ are initialy placed at clockwise consecutively numbered nodes of a regular polygon with $n-1$ nodes, the abstract team n is not placed in the polygon [3]. The team placed in the node on one side of the symmetric axis plays against its counterpart on the other side, and the abstract team placed on node 1 plays against team n. After each round, each abstract team is moved clockwise to the immediately next node until all $n-1$ assignments are completed. Figure 1 depicts how the polygon method is executed for 8 teams and Table 1 shows the results from the execution of the polygom method.

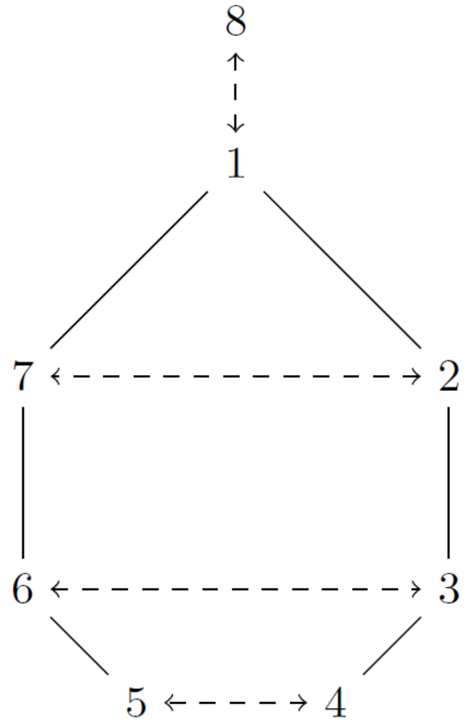


Figure 1: Polygon method for $n = 8$.

$T_i \setminus R_k$	1	2	3	4	5	6	7
1	8	6	4	2	7	5	3
2	7	5	3	1	6	4	8
3	6	4	2	7	5	8	1
4	5	3	1	6	8	2	7
5	4	2	7	8	3	1	6
6	3	1	8	4	2	7	5
7	2	8	5	3	1	6	4
8	1	7	6	5	4	3	2

Table 1. Results produced by the execution of polygon method.

The next stage in creating an initial schedule is assigning real teams to the abstract teams, the abstract teams that are consecutively played more times are assigned to real teams with smaller distances between their home cities. Then the final stage is to assign venues to each game, venues are assigned randomly. Table 2 below depicts how the final schedule should look like after initialization. Home and away games are represented by different signs, $-t_i$ is an away game and t_i is a home game.

$T_i \backslash R_k$	1	2	3	4	5	6	7
1	-8	-6	4	2	7	-5	3
2	7	5	3	-1	6	-4	-8
3	6	-4	-2	7	5	-8	1
4	5	-3	1	-6	-8	2	7
5	-4	2	7	-8	3	1	6
6	3	-1	8	-4	2	-7	-5
7	-2	8	-5	-3	1	6	4
8	1	-7	6	-5	4	-3	2

Table 2. Final schedule produced after initialization.

The schedule is then duplicated, and the venues are reversed to create the second half of the tournament, and thus a round robin tournament is produced.

4.2 Neighborhoods

In this paper four different neighborhoods are defined and explored by the ABC algorithm.

4.2.1 Home-away Swap

The move swaps a venue for a game between team T_i and T_j. If team T_i initialy plays at home against T_j at round r_k and away at T_j's home at round r_l, then in the solution obtained by this neighborhood, T_i will play away at T_j's home at round r_k and home at round r_l. Consider the schedule with $n = 6$ and $2(n-1)$ rounds. Below we show the appplication of home away swap to teams T_2 and T_5.

$T_i \backslash R_k$	1	2	3	4	5	6	7	8	9	10
1	-6	-4	2	5	-3	6	4	-2	-5	3
2	**5**	-3	1	-4	6	**-5**	3	-1	4	-6
3	-4	-2	-5	6	1	4	2	5	-6	-1
4	3	1	6	-2	5	-3	-1	-6	2	-5
5	**2**	6	-3	1	4	**-2**	-6	3	-1	-4
6	-1	5	-4	3	2	1	-5	4	-3	-2

Table 3. Schedule before Home Away Swap.

$T_i \backslash R_k$	1	2	3	4	5	6	7	8	9	10
1	-6	-4	2	5	-3	6	4	-2	-5	3
2	**-5**	-3	1	-4	6	**5**	3	-1	4	-6
3	-4	-2	-5	6	1	4	2	5	-6	-1
4	3	1	6	-2	5	-3	-1	-6	2	-5
5	**-2**	6	-3	1	4	**2**	-6	3	-1	-4
6	-1	5	-4	3	2	1	-5	4	-3	-2

Table 4. Schedule After Home Away Swap.

4.2.2 Team swap

This move selects two teams T_i and T_j randomly and swaps the schedule of the two teams except when they play against each other. The corresponding lines of the opponents of T_i and T_j must be changed as well *i.e* the rest of the schedule must be updated to produce a double round robin tournament. Below we show the application of team swap to teams T_3 and T_6.

$T_i \backslash R_k$	1	2	3	4	5	6	7	8	9	10
1	-6	-4	2	5	-3	6	4	-2	-5	3
2	5	-3	1	-4	6	-5	3	-1	4	-6
3	**-4**	**-2**	**-5**	6	**1**	**4**	**2**	**5**	-6	**-1**
4	3	1	6	-2	5	-3	-1	-6	2	-5
5	2	6	-3	1	4	-2	-6	3	-1	-4
6	**-1**	**5**	**-4**	3	**2**	**1**	**-5**	**4**	-3	**-2**

Table 5. Schedule before Team Swap.

$T_i \backslash R_k$	1	2	3	4	5	6	7	8	9	10
1	**-3**	-4	2	5	**-6**	**3**	4	-2	-5	**3**
2	5	**-6**	1	-4	**3**	-5	**6**	-1	4	**-3**
3	**-1**	**5**	**-4**	6	**2**	**1**	**-5**	**4**	-6	**-2**
4	**6**	1	**3**	-2	5	**-6**	-1	**-3**	2	-5
5	2	**3**	**-6**	1	4	-2	**-3**	**6**	-1	-4
6	**-4**	**-2**	**-5**	3	**1**	**4**	**2**	**5**	-3	**-1**

Table 6. Schedule after Team Swap.

4.2.3 Round Swap

This move selects two rounds randomly r_k and r_l and then simply swaps all the games between the two rounds.

4.2.4 Partial Round Swap

This move selects a random team T_i and two rounds r_k and r_l, and swaps T_i's games at these two rounds. The rest of the schedule is updated in order to produce a double round robin tournament.

4.3 Artificial Bee Colony(ABC)

Swarm intelligence [16] is a design framework based on social insect behaviour. Social insects such as bees and ants are unique in the way that these simple individuals cooperate to accomplish complex, difficult tasks. There is no centralized control, cooperation is distributed among the entire population. Each individual simply follows a set of rules influenced by locally available information [16]. This emergent behaviour results in great achievements that no single member could complete by themselves. Properties of swarm intelligent systems include robustness against individual misbehavior or loss, the flexibility to change quickly in a dynamic environment and an inherent distributed action.

ABC[17, 10] is an optimization algorithm based on the foraging behaviour of a honey bee swarm. The ABC algorithm starts with n solutions(food sources), the fitness of each solution is evaluated by a fitness function. The bees aim at discovering places of food sources with higher nectar(good fitness) [10]. If a new solution has fitness higher than the previous solution, the new position is updated and the previous one is forgotten.

The ABC algorithm has three phases: employed bee, onlooker bee and scout bee. In the employed bee and the onlooker bee phases, bees exploit the sources by local searches in the neighborhood of the solutions selected based on deterministic selection in the employed bee phase and the probabilistic selection in the onlooker bee phase. In the scout bee phase, solutions that are not beneficial for search progress are abandoned and new solutions are inserted instead of them to explore new regions in the search space later [17]. The algorithm has a well balanced exploration and exploitation ability.

In this paper the new solution v is obtained by exploring the different neighborhoods defined. The selection probability is given by:

$$prob_i = \frac{fitness(i)}{\sum_{i=1}^{n} fitness(i)}$$

Where n is the $colonySize/2$. Below is pseudo-code for ABC adapted from [10].

Algorithm 1: Artificial Bee Colony

```
indent=2em Parameters: $colonySize$, $limit$ and $maxCycle$. Initialize food sources
randomly. Evaluate the fitness of the population. $i = 0$ to $maxCycletimes$ $j = 0$ to
$colonySize/2$ Employed Bee phase Select $k$, $j$ and $r$ at random such that
$k \in \{1, ..., colonySize\}$,$j \in \{1, ..., d\}$ $r \in [0, 1]$ $v \leftarrow x_{ij} + r * (x_{ij} - x_{ik})$ //execute the
neighborhoods to get new solution. Evaluate solutions $v$ and $x_i$
$fitness(v) > fitness(x_i)$ Greedy selection $count_i \leftarrow count_i + 1$ $j = colonySize/2 + 1$ to
$colonySize$ Onlooker phase calculateSelectionProbability() select a bee using selection
probability $v \leftarrow x_{ij} + r * (x_{ij} - x_{ik})$ //execute the neighborhoods to get new solution.
Evaluate solutions $v$ and $x_i$ $fitness(v) > fitness(x_i)$ Greedy selection
$count_i \leftarrow count_i + 1$ $j = 1$ to $colonySize/2$ Scout phase $count_i > limit$ $x_i \leftarrow init()$
memorizeBestSolution()
```

5 Results

5.1 Data Set

Two data sets are used in this paper. The first data set is formed by circle instances artificially generated for testing whether TTP problems are easier when the associated traveling salesman instances are trivial [3]. The name of the instance is denoted by Circ$n$, where n is the number of teams and $8 \leq n \leq 16$. The second data set is formed by the National League(NL) instances which were generated by measuring the distances between the home cities of the teams participating in the league. The name of the instance is denoted by NLn, where n is the number of teams playing in the league and $8 \leq n \leq 16$.

5.2 Computational results and discussion

We performed experiments with the ABC algorithm, the colony size was set to 20, the number of cycles for foraging was also set to 20 and the algorithm was run 2 times with different starting solutions in order to see its robustness. Table 7 shows the best, and worst solutions obtained for each instance over 2 runs, as well as the computational time measured in seconds. Table 8 illustrates the performance of ABC compared to other previous results found in literature. Table 9 shows the comparisons of computational time for our algorithm with the algorithms we used for benchmarking (GRILS-mTTP and GA-SA). Running time for TTSA was not given in the original paper.TTSA[1] had the best known results at the time of the writing, the last column GAP is the relative gap in percentage between the values of the best known solution and the best solution found by our algorithm.

The ABC algorithm was very competitive on smaller instances ($n \leq 14$) and in some instances our proposed approach obtained results that are better than those of the best known solution. In instances Circ14, Circ12, NL8, NL14, the algorithm improved the best known solution values by 2.3%, 1.9%, 11.9% and 0.9% respectively. In most of the instances ($n \leq 14$),the results of ABC surpass the ones of GRILS_mTT [3] and GA-SA [2]. The algorithm did not perform well on larger instances($n > 14$) *e.g.* in Circ16 and NL16 the relative gap in percentage is well over 10. From table 9 we can observe that ABC shows good performance with regards to computational time compared to GA-SA on all NLn ($n \leq 14$) instances and only on two Circ$n$ ($n = 8$ and $n = 10$) instanes,and only outperformed GRILs-mTTP on instances Circ10, NL10 and NL16.

Instance	Worst	Best	Time(sec)
Circ8	139	129	17.637
Circ10	290	269	29.855
Circ12	450	415	65.635
Circ14	915	718	86.329
Circ16	1313	1227	312.636
NL8	41883	34973	18.289
NL10	71996	62972	27.446
NL12	125472	118533	45.081
NL14	191282	188547	82.398
NL16	348194	320420	198.276

Table 7. Computational results

Instance	TTSA	GRILS-mTTP	GA-SA	ABC	GAP%
Circ8	132	140	142	129	-2.3
Circ10	254	276	282	269	5.9
Circ12	420	456	458	415	-1.9
Circ14	682	714	714	718	5.2
Circ16	976	1004	1014	1227	25.7
NL8	39721	41928	43112	34973	-11.9
NL10	59583	63832	66264	62972	5,7
NL12	112298	120655	120981	118533	5,6
NL14	190056	208086	208086	188547	-0.8
NL16	267194	285614	290188	320420	19.9

Table 8. ABC results with two set of benchmark problems.

Instance	GRILS-mTTP	GA-SA	ABC
Circ8	1.4	48	17.637
Circ10	376.0	365	29.855
Circ12	8.5	51	65.635
Circ14	1.1	26	86.329
Circ16	115.3	264	312.636
NL8	0.7	55	18.289
NL10	643.9	130	27.446
NL12	24.0	317	45.081
NL14	69.9	140	82.398
NL16	514.2	142	198.276

Table 9. Comparison of computational time

6 Conclusion

In this paper, the traveling tournament problem is solved by an artificial bee colony algorithm. Four neighborhoods were defined and explored by the algorithm. Though the results obtained by the algorithm were promising, the algorithm did not perform well on larger instances. One direction for future work will be to define more neighborhoods with the aim of improving the performance of the algorithm on larger instances and in addition we intend to experiment more with the parameters of ABC, particularly the number of cycles for foraging and the colony size.

Bibliography

[1] A.Anagnostopoulos, L.Michel, P.Van Hentenryck and Y.Vergados. A Simulated Annealing Approach to the Traveling Tournament Problem. Brown University, 2005.

[2] F. Biajoli, L. Lorena. Mirrored Traveling Tournament Problem: An Evolutionary Approach. Lecture Notes in Computer Science,vol 4140,springer, pp. 208-217,2006.

[3] C.Ribeiro and S.Urrutia. Heuristics for the mirrored traveling tournament problem. Departement of Computer Science, Universidade Federal Fluminense, Brazil. School of Computer Science, Catholic University of Rio de Janeiro, Brazil, 2005.

[4] L.Gaspero and A.Schaerf. A Tabu Search Approach to the Traveling Tournament Problem. Dipartimento di Ingegneria Elettrica, Gestionale e Meccanica Universita di Udine, 2005.

[5] A.Goldbera. Highly Constrained Sports Scheduling with Genetic Algorithms, Department of Mathematics and Computer Science, Armhest College,2003.

[6] A.Guedes, C.Riberio. A hybrid heuristic for minimizing weighted carry-over effects in Round Robin tournaments, Instituto de Computacao, Universidade Federal Fluminense and Instituto de Computacao, Universidade Federal Fluminense, 2009.

[7] J.Hung, Y.Yen and K.Chein. Professional Sport Scheduling Optimization System Based on the Shortest Traveling Cost. Department of Computer Science and Information Engineering, Tamkang University, 2010.

[8] C.Juan and R.Razamin and I.Haslinda. A hybrid constraint-based programming approach to design a sports tournament scheduling. European Journal of Scientific Research, 49 (1). pp. 39-48. ISSN 1450-216X, 2011.

[9] A.Lim, B.Rodrigues and X.Zhang. A Simulated Annealing and Hill-climbing algorithm for The Traveling Tournament Problem. Department of IEEM, Hong Kong University of Science and Technology, Clearwater Bay, Kowloon, Hong Kong. School of Business, Singapore Management University, 2003.

[10] R.Parpinelli and H.Lopes. New inspirations in swarm intelligence: a survey. Bioinformatics Laboratory, Federal University of Technology Paran (UTFPR), Curitiba (PR), 80230-901, Brazil and Applied Cognitive Computing Group, Santa Catarina State University (UDESC), Joinville (SC), 89223-100, Brazil. Bioinformatics Laboratory, Federal University of Technology Paran (UTFPR), Curitiba (PR), 80230-901, Brazil, 2011.

[11] W.Wei, S.Fujimura, X.Wei and C.Ding. A Hybrid Local Search Approach in solving the Mirrored Traveling Tournament Problem. Graduate School of Information, Production and System, Waseda University, Kitakyasya, Japan.School of Electronic, Information and Electrical Enginnering, Shangai Jia Tong University, Shangai, China, 2010.

[12] D.Uthus. Sports Scheduling: An Artificial Intelligence Approach, University of Auckland, 2010.

[13] S.Young. Alternative Aspects of Sports Scheduling, 2004.

[14] Wikipedia. http://en.wikipedia.org/wiki/Combinatorial_optimization. Accessed on February 2012.

[15] Wikipedia. http://en.wikipedia.org/wiki/Round-robin_tournament.
Accessed on March 2012

[16] Swarm Intelligence http://www.bluetronix/Swarm_Intelligence.html. Accessed on March 2012

[17] Wikipedia. http://en.wikipedia.org/wiki/Artificial_bee_colony_algorithm. Accessed on March 2012

Proceedings of the 2012 ORSSA Annual Conference
pp. 20–30
www.orssa.org.za/wiki/uploads/Conf/2012ORSSAConferenceProceedings.pdf

ORSSA Proceedings
ISBN
978-07972-1406-4
© 2012

Artificial Neural Networks to detect Risk of Type 2 Diabetes

BY Baha[1] AO Adewumi[2] NV Blamah[3] GM Wajiga[4]

Abstract

This paper proposes a new model for identifying individuals at risk of developing type 2 diabetes (T2D). This study aims at early identification of individuals at risk of T2D with data obtained from Nigeria. The overall goal is to contribute to government efforts in decreasing morbidity and mortality rate linked to T2D as well as to assist stakeholder both in taking preventive measure and for future forecast. Data was collected from three different sources which include both professional and non-professional respondents. An Analytical hierarchy process (AHP) was first applied on the dataset. The result from the AHP was then used to proffer a new ANN model that could identify individuals at risk of developing T2D. The best performed network identified during the training consisted of 2 hidden layers of 6 and 2 neurons and an output layer of 1 neuron. The network recorded the best validation performance of 0.1000 at 532th epoch and correlation coefficient of 0.9981. A regression plot indicates exact linear relationships with all the axes close to 1. At least 266 out of 1122 of the dataset used were found close to 1, which indicated high risk of T2D.

Key words: Type 2 diabetes, Risk factors, artificial neural network, Matlab, Backpropagation.

1 Background

Diabetes is a chronic lifelong disease characterized by hyperglycaemia resulting from defects in insulin secretion, insulin action, or both; which increases the risks of long-term damage, dysfunction and failure of various organs especially the eyes, kidneys, nerves, heart, gallbladder or blood vessels. Type 2 diabetes (T2D) occurs due to insufficient insulin secretion by beta cells or development of insulin resistance, where the cells of the body do not accept the insulin. It constitutes 85 – 90% of all cases of diabetes and usually occurs in adults over 40 years of age [1]. Excess global mortality attributable to T2D in the year 2000 was 1 million deaths in developing nations and 1.9 million deaths in developed nations, or 2.8% of all deaths globally [2]. Our major motivation for this

[1] Information Technology & Systems, Mainstreet Bank, Nigeria, email: bybaha@yahoo.com
[2] Corresponding author: University of KwaZulu-Natal, South Africa, email: adewumia@ukzn.ac.za
[3] University of KwaZulu-Natal, South Africa, email: blamah@ukzn.ac.za
[4] Modibbo Adama University of Technology, Nigeria, email: gwajiga@gmail.com

research is the increasing need for early identification of individuals at high risk of developing T2D that could help decrease the incidence of morbidity and mortality. In order to achieve accurate identification, we first used an Analytical hierarchy process (AHP) technique to determine the strength of risk factors associated with development of T2D. The result obtained from the AHP technique was used in developing the model. This was then used to proffer a new artificial neural network (ANN) model that could identify individuals at risk of developing T2D.

2 Background and related works

Many epidemiologists have associated risk factors to the development of T2D. Højbjerre *et al.* [3] reported physical inactivity as a strong risk factor for developing T2D that may be more detrimental in first-degree relative subject, unmasking the underlying defects of metabolism. Katzmarzyk [4] also outlined most current physical activity guidelines on achieving 30 minutes per day or 150 minutes per week of moderate–to–vigorous physical activity. Baecke *et al.* [5] developed a questionnaire for evaluating a person's physical activity into three distinct dimensions namely, work, sports and leisure. Kaprio *et al.* [6] estimated the contribution of heritable (rather environmental factors) factor to T2D to be as high as 40% or more. Arslanian *et al.* [7] reported that a family history is positive if there is a family member in the immediate three generations (siblings, parents, or grandparents) with known T2D. Mykkanen *et al.* [8] also reported that obesity, central obesity and a family history of diabetes are significantly associated with the increased prevalence of impaired glucose tolerance and noninsulin-dependent diabetes mellitus, especially in elderly subjects. According to Wing *et al.* [9], children of obese parents are 10 times more likely to be obese than children with parents of normal weight.

In elderly people, the risk of death from diabetes is greater than the sum of all accidents and other external causes [10]. Kwon *et al.* [11] examined the effects of age, period, and birth cohort on the prevalence of diabetes and obesity in Korean men. Results obtained from the study indicated that age is a relatively important predictor for the prevalence of diabetes with an increasing trend in younger birth cohorts. Various studies have shown differences by sex in the effects of endogenous sex hormones on insulin resistance [12, 13]. In men, low plasma testosterone is associated with obesity, upper body fat distribution and increased levels of glucose and insulin, whereas hyper-androgencity is associated with an increased risk for T2D and cardiovascular disease in women [12]. In Thorand *et al.* [13], the effect of modification by sex in the association between inflammation and T2D was studied. Result from the study indicated stronger associations in women than men. Khan *et al.*]14] reported heavy influence of ethnicity and sex on diabetes incidence and outcomes in major Asian subgroups. According to Thanopoulou *et al* [15], increased animal fat in the diet may contribute to increased incidence of diabetes. Lapidus *et al.* [16] observed association between alcohol and diabetes as well as mortality, which is dependent on a number of confounding factors, most importantly Body Mass Index. It has also been noted that sleep duration is a risk factor to the development of diabetes in middle-aged and elderly men [17]. The study posited that sleep duration is short if it is less than or equal to five or six hours per night and long if it is greater than eight hours per night.

Jaafar and Ali [18] designed an ANN model of 8 inputs, 3 hidden layers and 1 output layer to detect individuals with diabetes mellitus. The result indicated 268 out of 768 patients as suffering from diabetes. Similarly, Rao *et al.* [19] developed and trained clinical decision support system using ANN to predict the well-being of individuals with diabetes using a secondary data of 241 diabetic patients. The study implemented a multilayer perceptron with the Java Swing Package using JDK 1.5. Neural network can be used to estimate posterior probabilities directly and distribution-free unlike traditional statistical procedures such as discriminant analysis and logistics regression [20].

Another study has reported the use of ANN for the preprocessing and diagnosis of diabetes mellitus, which was useful in classifying T2D of Pima Indian Diabetes data set [21]. The study combined two methods, *replace with mean* and *replace with k–nearest neighbor,* with principal component analysis for preprocessing. Result from the study show some improvement in the accuracy of the prediction of T2D.

3 Measuring severity of risk factors using AHP

In order to determine the relative importance of alternatives in decision-making, Saaty [22] developed an AHP in which ideas, feelings, and emotions affecting decision process are quantified using pair-wise comparison. AHP uses a fundamental scale of absolute numbers that are proved by practice and validated by decision problem experiments [23]. AHP does not only help to structure group discussions, it also supports quantitative analyses by generating inconsistency ratio, and weighting factors that can be visualized in graphs [24]. The resultant weight is then compared and ranked with other alternatives, which can assist decision makers in making choices. A framework that shows the stages involved in using AHP for feature selection is presented in Figure 1.

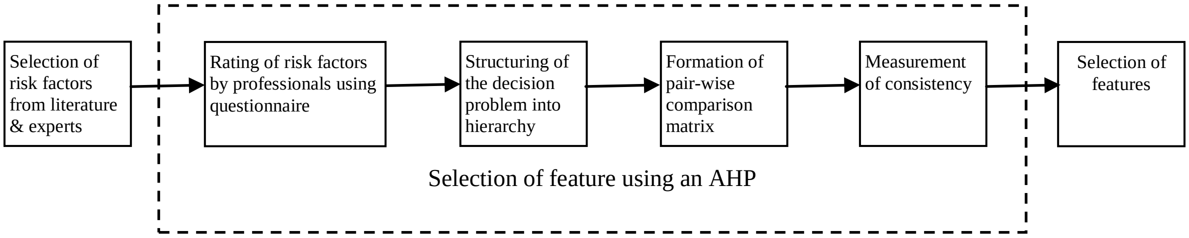

Selection of risk factors from literature & experts
Rating of risk factors by professionals using questionnaire
Structuring of the decision problem into hierarchy
Formation of pair-wise comparison matrix
Measurement of consistency
Selection of features
Selection of feature using an AHP

Figure 1: *Framework for selection of feature (Source: [25])*

In this study, data was collected from three different sources. The first set was obtained from epidemiological studies with the help of Diabetologist in which 13 risk factors associated with development of T2D were identified. The second source came from professional respondents with the use of structured likert formatted questionnaires to ascertain degree of severity of risk factors. The last source of data, which was used for training, validation and testing of the network, came from non-professional respondents. In order to rate the risk factors, epidemiological studies and diabetologist were used in identifying 13 risk factors namely, heredity, physical inactivity, age, obesity, dietary, smoking, alcohol, impaired glucose tolerance (IGT), gestational diabetes, close marital alliance, ethnicity, duration of sleep, and sex. The views of 408 professionals were collected using structured likert format with five choices to ascertain the degree of severity of the risk factors. AHP technique was then applied to select the most contributing risk factors in the development of T2D. Result obtained from the AHP revealed that heredity factor contributes 0.5388, obesity 0.1038, physical inactivity 0.0230, dietary 0.0230, age 0.1038, IGT 0.1038 and gestational diabetes 0.1038.

4 Neural network design

The Neural Network Toolbox of MATLAB 7.10.1 (R2010a) was used to design the neural network for this study. The toolbox is a set of functions and structures that handle neural networks such that writing complex code for activation functions, configuring network, initializing weights, training algorithms, preprocessing and post-training analysis are not required. A matrix of 6 risk factors of T2D variables corresponding to 1122 different respondents together with relative valuations were used as input and target data. The first subset was training set, which computes the gradient, update network weights and biases. This constitutes 70% of dataset collected from respondents. The second subset was validation set, which represents 20% of dataset while the remaining 10% of the dataset formed the test set. The first layer of the network has weights coming from the input matrix of 6 variables while each subsequent layer has weight coming from the previous layer. The input layer of the toolbox is not always used in literature [26]. Figure 2 summarizes the overall workflow for the design process used in this study.

4.1 Network training

The neural network was trained using batch mode backpropagation algorithm with gradient descent and momentum. The number of neurons at the hidden layers was varied, as well as the number of epochs, learning rate and momentum coefficient. These parameters were varied to improve performance of the neural network. Ten best results obtained from the training are detailed in Table 1.

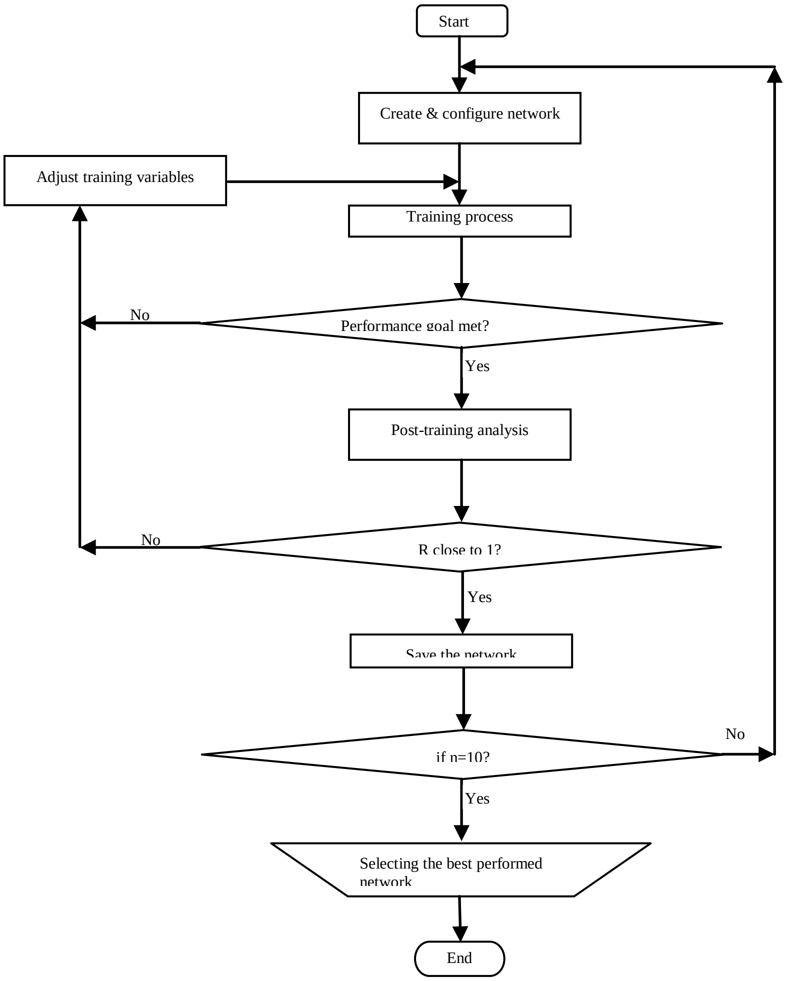


Figure 2: *Design layer of the Neural Network used in this study*

Table 1: *Ten best Network Training Results*

SNO	Network configuration	No of epochs	Time	Training Performance	Gradient	Validation Performance	Correlation coefficient
1	2,1	124	7	0.0994	0.0392	0.0989	0.9789
2	3,1	177	7	0.0996	0.0327	0.1038	0.9072
3	4,1	178	6	0.1000	0.0299	0.0999	0.9920
4	6,1,1	359	13	0.1000	0.0241	0.1000	0.9917
5	6,2,1	532	21	0.1000	0.0224	0.1000	0.9981
6	6,3,1	551	24	0.0999	0.0215	0.0999	0.9925
7	6,4,1	698	33	0.1000	0.0203	0.1000	0.9957
8	6,5,1	857	34	0.0999	0.0190	0.0998	0.9957
9	6,6,1	899	35	0.0999	0.0184	0.0999	0.9976
10	6,7,1	943	37	0.0998	0.0184	0.0999	0.9983

Usually, the network training performance is evaluated by an error computation which determines the variance of the network's output from the ideal output. Performance plots are used to indicate that the iterations at which the validation performance reached a minimum are the same as iterations at which the trainings were stopped, that is, best iteration and maximum iteration are equal in all the performance plots. Furthermore, all curves from this study indicate no major problems with the training. All the training, validation and test curves are very similar. This implies that there was no overfitting and that the chosen model fits the data correctly without poor description of the underlying data-generating process. Another factor in performance evaluation is regression analysis. In the regression plots, all the three axes represent the training, validation and test data. The dashed lines represent the difference between the ideal result and the network output. The solid lines represent the best fit linear regression line between network outputs and network targets. The correlation coefficient is an indication of the relationship between the outputs and the targets. Typically, if correlation coefficient is close to 1, it indicates an exact relationship between outputs and targets. If it is close to 0, then there is no linear relationship between the outputs and the targets. In all the regression plots, the network outputs and targets were almost exactly equal and the value of correlation coefficient for training, validation and test were all close to 1. This implies that linear relationships are almost perfect and that the networks were well trained. Figure 3 depicts correlation coefficients of the ten network configurations obtained during network training.

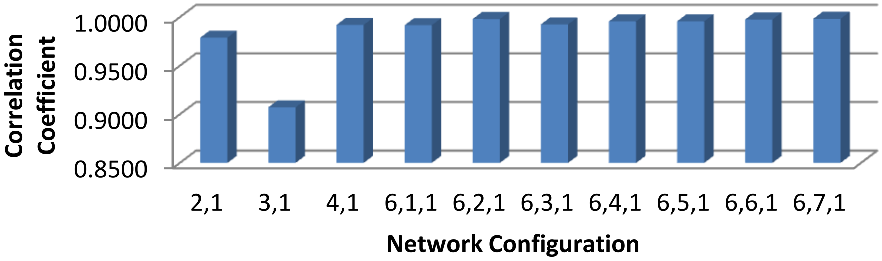


Figure 3: *Correlation Coefficients for Ten Network Configurations*

A further analysis of the training performance, validation performance and correlation coefficient in this study reveal that 4 network configurations recorded 0 as training performance error. These are 4-1, 6-1-1, 6-2-1 and 6-4-1. Similarly, network with configurations 6-1-1, 6-2-1, and 6-4-1 recorded 0 as validation performance error. In terms of correlation coefficient error, none of the network configurations recorded 0. However 6-7-1, 6-2-1 and 6-6-1 recorded the least three correlation coefficient errors as 0.0017, 0.0019 and 0.0024 respectively. This indicates that correlation coefficient of 6-7-1 is closest to 1, followed by 6-2-1, 6-6-1 and so forth. Since network training is only acceptable if the error calculation is low [27] and networks are best generalized at the minimum of the validation error [28], 6-2-1 was adjudged the best network configuration because it recorded 0 in training and validation performance. In addition, it has the least total error as indicated in Table 2.

Thus, a feedforward neural network of 6-2-1 with correlation coefficient of 0.9981 and validation performance of 0.1, which occurred within 21 seconds at epoch 532 was adjudged the best network configuration in this study. The network used dot product weight functions, net sum input functions, log-sigmoid transfer function at hidden layers and linear transfer function at output layer. Figure 4 represents ANN architecture for neural network 6-2-1. The training indicated 266 out of 1122 cases close to 1, and the rest close to 0.

Table 2: *Error Factors of Training, Validation Performance and Correlation Coefficient*

Network configuration	Training performance Error	Validation performance Error	Correlation Coefficient Error	Total Error
2,1	0.0006	0.0011	0.0211	0.0228
3,1	0.0004	-0.0038	0.0928	0.0894
4,1	0.0000	0.0001	0.0080	0.1122
6,1,1	0.0000	0.0000	0.0083	0.0083
6,2,1	0.0000	0.0000	0.0019	0.0019
6,3,1	0.0001	0.0001	0.0075	0.0102
6,4,1	0.0000	0.0000	0.0043	0.0043
6,5,1	0.0001	0.0002	0.0043	0.0046
6,6,1	0.0001	0.0001	0.0024	0.0089
6,7,1	0.0002	0.0001	0.0017	0.0020

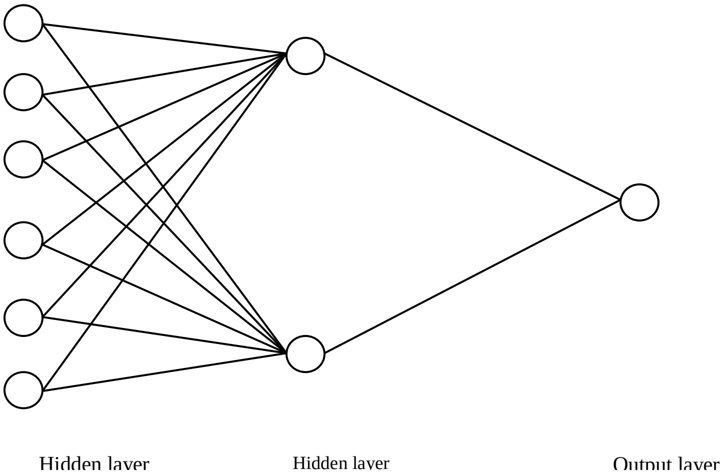


Figure 4: *ANN Architecture selected for this study.*

If x_1, x_2, x_3, x_4, x_5 and x_6 represent inputs of heredity, physical inactivity, dietary, Age, IGT and gestational diabetes respectively and w_1, w_2, w_3, w_4, w_5 and w_6 are connecting weights between the hidden layers, then mathematically we have:

i. Input to the second hidden layer is

$$H_{in} = \sum_{n=1}^{6} x_n w_n \dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots .1$$

ii. Output from the second hidden layer is

$$H_{out} = f(H_{in}) = f\left(\sum_{n=1}^{6} x_n w_n\right) \dots\dots\dots\dots\dots\dots\dots\dots\dots\dots ..2$$

iii. Input to the output layer is

$$y_{in} = \sum_{n=1}^{2} (H_{out})_n \dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots ..3$$

iv. Output from the output layer is

$$y_{out} = f^{'}(y_{in}) \dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots ..4$$

where *f* and *f'* sigmoid and linear activation functions respectively. H_{in} and H_{out} are input and output at the second hidden layer, while y_{in} and y_{out} are input and output at the output layer.

5 Conclusion

The best ANN structure for identifying individual at risk of developing T2D was selected using correlation coefficient, validation performance, training performance, time taken to reach the performance goal, gradient and number of epochs. Figure 5 depicts performance plot for neural network 6-2-1. This recorded a training performance of 0.1 in 21 seconds and validation performance of 0.10004, which occurred at epoch 532.

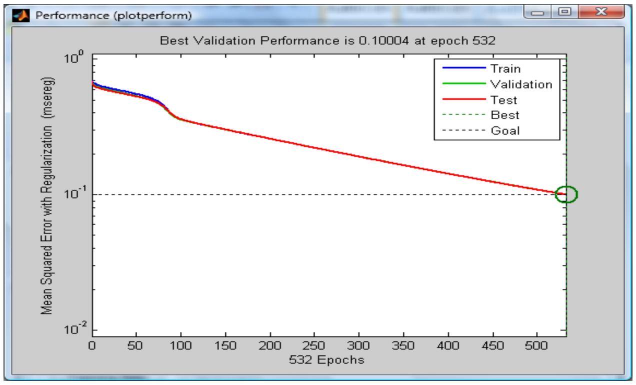


Figure 5: *Performance validation during training*

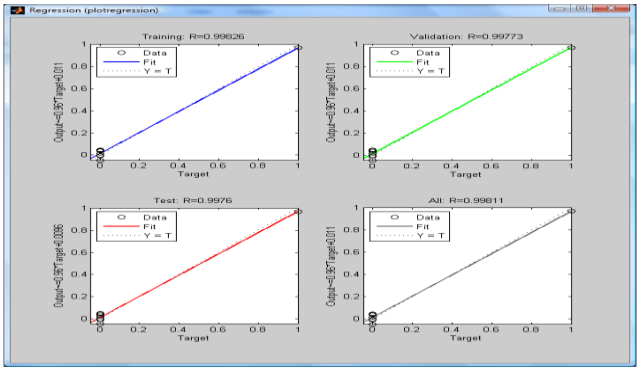


Figure 6: *Regression analysis obtained from post-training analysis*

A regression plot between outputs of the network and targets is in Figure 6. The dashed lines represent targets, which is the difference between perfect result and network outputs. The solid lines represent the best fit linear regression line between network outputs and targets. The correlation coefficients between outputs and targets in all the axes are close to 1, which indicates exact linear relationships that the network has been well trained. The overall correlation coefficient obtained from the regression analysis is 0.99811. The figure depicts similar curves in training, validation, and testing, which indicates that the network has been well trained and could be used for identifying individuals at high risk of T2D.

In conclusion, the study has presented a multilayer feedforward backpropagation network with two hidden layers of 6 and 2 neurons and an output layer of 1 neuron. A total of 266 out of 1122 cases were found close to 1, which indicates high risk of developing T2D. This suggests the need to make changes in dietary behavior and/or participate actively in physical exercise. Further studies could use this design for web-based program that will assist in identifying individuals at high risk of developing T2D.

Bibliography

[1] Boutayeb, A., Twizell, E. H., Achouayb, K. & Chetouani, A. (2004). A mathematical model for the burden of diabetes and its complications. *BioMedical Engineering OnLine*, *3*(20). http://www.biomedical-engineering-online .com/content/3/1/20

[2] Franks, P. W., Hanson, R. L., Knowler, W. C., Moffett, C., Enos, G., Infante, A. M., Krakoff, J. & Looker, H. C. (2007). Childhood Predictors of young-onset Type 2 diabetes. *Diabetes, 56,* 2964 – 2972.

[3] Højbjerre, L., Sonne, M. P., Alibegovic, A. C., Dela, F., Vaag, A., Bruun, J. M., Christensen, K. B. and Stallknecht, B. (2010). Impact of physical inactivity on subcutaneous adipose tissue metabolism in healthy young male offspring of patients with Type 2 diabetes. *Diabetes*, *59*, 2790–2798.

[4] Katzmarzyk, P. T. (2010). Physical Activity, Sedentary Behavior, and Health: Paradigm Paralysis or Paradigm Shift. *Diabetes, 59,* 2717 – 2725

[5] Baecke, J.A.H., Burema, J. and Frijters, E.R. (1982). A short questionnaire for the measurement of habitual physical activity in epidemiological studies. *Am J ClinNutr, 36,* 936-942. www.cebp.nl/vaultpublic/filessystem/?ID =1247

[6] Kaprio, J., Tuomilehto, J. and Koskenvuo, M. (1992). Concordance for type 1 (insulin dependent) and Type 2 (non-insulin-dependent) diabetes mellitus in a population-based cohort of twins in Finland. *Diabetologia, 35,* 1060–1067.

[7] Arslanian, S.A., Bacha, F., Saad, R. and Gungor, N. (2005). Family history of Type 2 diabetes is associated with decreased insulin sensitivity and an impaired balance between insulin sensitivity and insulin secretion in white youth. *Diabetes care*, *28,* 127–131.

[8] Mykkanen, L., Laakso, M., Uusitupa, M. and Pyorala K. (1990). Prevalence of diabetes and impaired glucose tolerance in elderly subjects and their association with obesity and family history of diabetes. *Diabetes care, 13,* 1099–1105

[9] Wing, R. R., Venditti, E., Jakicic, J. M., Polley, B. A. and Lang, W. (1998). Lifestyle intervention in overweight individuals with a family history of diabetes. *Diabetes care, 21*, (3), 57–64.

[10] Peláez, M. and Vega, E. (2006).Old age, poverty and the chronic disease epidemic in Latin America and the Caribbean. *Diabetes Voice, 51*(4), 30 – 33.

[11] Kwon, J., Song, Y., Park, H. S., Sung, J., Kim, H. and Cho, S. (2008). Effects of age, Time period, and birth cohort on the prevalence of diabetes and obesity in Korean men. *Diabetes Care, 3*(2), 1255–260.

[12] Oh, J., Barrett-Connor, E., Wedick, N. M. and Wingard, D. L. (2002). Endogenous sex Hormones and the development of Type 2 diabetes in older men and women: the rancho Bernardo study. *Diabetes Care,25*, 55–60.

[13] Thorand, B., Baumert, J., Kolb, H., Meisinger, C., Chambless, L., Koenig, W. and Herder, C. (2007). Sex Differences in the Prediction of Type 2 Diabetes by Inflammatory Markers. *Diabetes Care, 30*(4), 854–860.

[14] Khan, N. A., Wang, H., Anand, S., Jin, Y., Campbell, N. R.C., Pilote, L. and Quan, H. (2011). Ethnicity and sex affect diabetes incidence and outcomes. *Diabetes Care, 34*, 96-101.

[15] Thanopoulou, A.C., Karamanos, B.G., Angelico, F. V., Assaad-Khalil, S. H., Barbato, A. F., Ben, M. P. D., Djordjevic, P. B., Dimitrijevic-Sreckovic, V. S., Gallotti, C. A., Katsilambros, N. L., Migdalis, I. N., Mrabet, M.M., Petkova, M. K., Roussi, D. P. and Tenconi, M. P.(2003). Dietary Fat Intake as Risk Factor for the Development of Diabetes. *Diabetes care, 26*(2), 234 – 240.

[16] Lapidus, L., Bengtsson, C., Bergfors, E., Bjorkelund, C., Spak, F. and Lissner, L. (2005). Alcohol intake among women and its relationship to diabetes incidence and all-cause mortality: the 32-year follow-up of a population study of women in Gothenburg, Sweden. *Diabetes Care, 282230-2235. http://care.diabetes journals.org/content/28/9/2230.full.pdf +html?sid=ca41ea24-52bf-4475-a915-eb83b920f6af*

[17] Yaggi, H. K., Araujo, A. B. andMckinlay, J. B. (2006). Sleep duration as a risk factor for the development of Type 2 diabetes. *Diabetes Care, 29*(3), 657–661.

[18] Jaafar, S. F. & Ali, D. M. (2005). Diabetes Mellitus forecast using Artificial Neural Network. Paper presented at the Asian Conference on Sensors and the International Conference on New Techniques in Pharmaceutical and Biomedical Research processing. http:/www.computersociety/ieee/

[19] Rao, A. P., Rao, N., Manda, R., Sridhar, G. R., Madhu, K., Veena, S., Madhavi. R., Sangeetha, B. S., Rani, A. & Hanuman, T. (2002). A clinical decision support system using artificial neural network to assess well being in diabetes, Unpublished manuscript, Endocrine and Diabetes centre, Krishnanagar, Visakhapatnam, India, 15-12-16. http://www. allamopparao.org/en/papers/paper10.pdf

[20] Shanker, M., Hu, M. Y. & Hung, M. S. (1999). Estimating probabilities of diabetes mellitus using Neural Networks. http:/personal.kent.edu/¬mshanker/personal/zip˙files/sar˙2000. pdf

[21] Jayalskshmi, T. & Santhakumaran, A. (2010). Impact of Preprocessing for Diagnosis of Diabetes Mellitus Using Artificial Neural networks. Paper presented at Second International Conference on Machine Learning and Computing, IEEE Computer Society, 109 – 112.

[22] Saaty, T. L. (1980). *The Analytic Hierarchy Process: Planning, Priority Setting, Resource Allocation.* McGraw-Hill, New York.

[23] Abdullahi, L. and Azman, F. N. (2011).Weights of obesity factors using analytic hierarchy process. *IJRRAS, 7* (1), 57 – 63. www.arpapress.com/Volumes/Vol7Issue1/IJRRAS˙7˙1˙ 09.pdf

[24] Saaty, T. L. (1999). Decision Making for Leaders: The Analytical Hierarchy Process for Decisions in a complex world. In Mukherjee, B., Das, P. (2010). The use of the Analytical Hierarchy Process as a tool for selection of important factors for the multi – disciplinary evaluation of medical devices. *International Journal of Academic Research, 2, 37 – 42.* www.ijar.lit.az

[25] Kumar, S., Parashar, N. and Haleem, A. (2009). Analytical Hierarchy Process applied to vendor selection problem:Small scale, medium scale and large scale industries. *Business Intelligence Journal 2*(2), 355–362.

[26] Beale, M. H., Hagan, M. T., & Demuth, H. B. (2011). Neural Network Toolbox^{TM} User's Guide. The Mathworks Inc. www.mathworks.com/

[27] Heaton, J. (2005). *Introduction to Neural Networks with Java, second edition.* Heaton Research. http://www.freetechbooks.com/introduction-to-neuralnetworks with-java-t639.html

[28] Huang, G., Saratchandran, P. and Sundararajan, N. (2005). A generalized growing and pruning RBF neural network for function approximation. *IEEE Transactions on Neural Networks, 16,* 57 – 67.

Proceedings of the 2012 ORSSA Annual Conference
pp. 40–47
www.orssa.org.za/wiki/uploads/Conf/2012ORSSAConferenceProceedings.pdf

ORSSA Proceedings
ISBN
978-07972-1406-4
©2012

Consumer Loan Decisions with Profit-Loss Tradeoff under Multiple Economic Conditions

K Rajaratnam[*] C Huang[†]

Abstract

A topic of recent interest is loss prevention and cash-flow management in consumer loan portfolios. Past literature considers the case of a portfolio manager with multiple but conflicting objectives, such as maximizing profits, maximizing market share, and minimizing risk. This was later extended to incorporate notions of economic conditions where a portfolio manager is faced with making a decision prior to the realization of future economic conditions. For example, the portfolio manager is faced with trade-off between maximizing expected profit and market share, under the possibility of one out of two future economic conditions. However, the accept/reject decision must be made prior to the realization of an economic condition. In this paper, we extend the notion of multiple economic conditions to the expected profit and expected loss space. We limit our discussion to a portfolio manager with access to a single scorecard, but with customer performance differing under two different economic conditions. We show all operating points on the efficient frontier is a result of single cutoff-score policy.

Key words: Portfolio Optimization, Banking, Decision Analysis

1 Background and Literature Review

Consumer loan portfolio managers typically use credit scorecards to evaluate loan applications. These scorecards map each applicant's personal data to a real valued output. These personal data may consist of application data, credit bureau information, demographic information and other data pertaining to an applicant that indicate risk behavior. The real value output, also known as a credit score, has a bijective relationship with probability of default. Typically, these scorecards forecast probability of an applicant defaulting within a specified window period, such as a year. Given such a score, various business metrics may be determined for each applicant with these customer business metrics aggregated into portfolio business metrics. Using these metrics, an accept/reject decision on the loan application is made. In South Africa, financial institutions have been using scorecards as part of the decision making process

[*] Corresponding author: Department of Finance and Tax, University of Cape Town, Private Bag, Rondebosch 7701, South Africa, email: kanshukan.rajaratnam@uct.ac.za.

[†] Department of Finance and Tax, University of Cape Town, Private Bag, Rondebosch 7701, South Africa, email: chun-sung.huang@uct.ac.za.

for secured and unsecured loans (e.g., home loans and credit cards). When making these decisions, portfolio managers typically maximize expected profit, maximize expected market share and minimize losses[1]. Generally, scorecards are built without considering economic cycles and hence the default probability forecasts do not consider future economic conditions. In other words, with current scorecard building process, management's expectations of default probabilities is independent of economic conditions.

Consider the following: during the credit crisis of 2007/2008, mortgage defaults increased drastically in the US. In turn, due to lower house demands, house prices dropped resulting in difficulties for consumers to re-finance their mortgages, causing further defaults. In addition, increase in unemployment rates resulted in further increase in mortgage defaults. These increases in defaults was not predicted by credit scores. When the economic condition worsened, so did the probability of default. It is intuitive that default probabilities be conditioned on the prevailing macroeconomic conditions and as such, scores should have a unique bijective relationship to default probabilities for each macroeconomic scenario.

Initial work on scoring decisions involved maximizing expected profit using a scorecard to evaluate the riskiness of each applicant ([2], [5], [6], [11]). Oliver and Wells studied the tradeoff between multiple objectives for a portfolio manager with a single scorecard [7]. See Beling *et al.* [1], Rajaratnam *et al.* [9] and Rajaratnam *et al.* [10] for further extensions.

A parallel related research is scorecard development. Logistic regression is typically used to construct a scorecard, but in the past decade more sophisticated methods and combination of methods have been employed in predicting probability of default through credit scores (see Hand and Henley [4] for a summary on scorecards pertaining to consumer credit decisions). Flat maximum effect shows there is little improvement in performance between different statistical models [8]. However, predictive power may be improved through new variables. Recently, researchers and practitioners have investigated combining economic variables into scorecards. There are numerous examples in literature where economic conditions affect scores and behaviour (see Zandi [12]; De Andrade and Silva [3]). Typically, scorecards are built with one dataset and then tested using another. Suppose the test dataset is separated into time periods of different economic conditions. For example, one may separate the test dataset into two, where one dataset contains customer's performance information during good economic conditions and another during bad economic conditions. The scorecard may then be tested for performance under these economic conditions. Given such a scenario, the portfolio manager may score each potential applicant using the scorecard. This results in a single score with two different bijective relationship between score and default probabilities under each economic condition.

Rajaratnam *et al.* [9] dealt with the case of a portfolio manager with access to a scorecard with different performance in different economic scenarios. However, the decision had to be made prior to account performance and hence before the future economic condition is revealed. This work showed the optimal policy is a single cutoff score policy when the portfolio manager is faced with a dual objectives of maximizing expected profit and market share. Given a cutoff score, the expected volume is independent of the future economic condition (after all, the account volume is determined by the cutoff score, independently of the future economic

[1] Portfolio managers are primarily concerned about default loss amount. In this paper, we use the terms “defaults losses” and “losses” interchangeably.

condition). However, in this paper, the portfolio manager is operating on the expected profit-expected loss (EPL) space and both these metrics are depended on the economic condition during account performance stage. Thus, we consider expected revenue and expected loss separately and construct the unconditional curve in these metrics with respect to expected volume before combining the two to determine the EPL curve. Further, we assume a portfolio manager has access to a single scorecard, and probabilities associated with the realization of economic conditions during future account performance period. We assume two mutually exclusive and collectively exhaustive economic conditions in this work, but is easily extendable to more economic conditions. The portfolio manager has a pool of applicants and must decide whether to offer/reject credit for each application. We construct the set of efficient operating points for the portfolio manager operating under the objectives of maximizing profit and minimizing loss.

In the following section, we introduce statistical notions of scores and scorecards; business metrics such as profit and loss given a score; and the concept of dominance in decision spaces. In Section 3, we consider the decision faced by a portfolio manager with the above mentioned scenario. Finally in Section 4, we summarize and discuss our findings.

2 Business Models

In this section, we introduce notation for credit scores, construct the basic models describing the bank's objectives, and define notion of efficient frontier.

2.1 Credit Scores and Receiver Operating Curve

Suppose information vector observed for a single applicant is the input to the scorecard. This information vector contains application and behavioral information for that customer. The scorecard output is a real-valued score, $s()$ used to forecast customer's performance over a given period. Additionally, we assume the performance outcome for each accounts is random variable Y which consist of two mutually exclusive and collectively exhaustive outcomes, G and B. The outcome G denotes an outcome of a customer not defaulting during the performance period. Similarly, B denotes an outcome for a defaulting (or *bad*) customer.

In this paper, we assume set of scores has a bijective relationship with probability of default, $p(B|s)$.[2] Similarly, given s, we may determine the probability of not defaulting $p(G|s)$. The prior belief that an applicant is good or bad is denoted by p_G and p_B. Further, let $f(s|G)$ and $f(s|B)$ denote the conditional likelihood of data s, and $f(s)$ the unconditional likelihood of data s. Thus, we can relate the likelihood of data s conditioned on the outcome G with the conditional probability of that score using Bayes Theorem, i.e., $p(G|s) = f(s|G)p_G/f(s)$. Similarly, $p(B|s) = f(s|B)p_B/f(s)$. We denote the conditional cummulative likelihood functions with $F(s|B)$ and $F(s|G)$, and the unconditional cummulative likelihood function with $F(s)$.

[2] Following Rajaratnam *et al.* [9], we assume default risk decreases with increasing score, i.e., $p(B|s)$ decreases with respect to increase in s.

2.2 Objective functions

A portfolio manager has multiple objectives in creating a portfolio, such as maximizing profit, maximizing market share, or minimizing default losses. We assume the bank earns $1 + r_L$ for each unit of loan given to a good account, where r_L is the return on loans. In order to lend a unit of credit, the bank needs to fund the loan amount. We assume loan volume is funded through debt at a rate of c_D, with $c_D < r_L$. For each unit of loan, the net-revenue from a good applicant is the return on the loan minus the cost of debt, i.e., $(r_L - c_D)$. Given a default event, the bank loses $C(l_D)$ for each unit of defaulted loan where C is the exposure at default and l_D is the fractional loss given default. Throughout this paper, we assume $C = 1$. Therefore, the net-loss of a bad account with score s is $(l_D + c_D)$.

In order to construct the portfolio level metrics, we denote portfolio expected revenue, loss, profit and volume as $E_S[R(s_c)]$, $E_S[L(s_c)]$, $E_S[P(s_c)]$, and $E_S[V(s_c)]$ under single cutoff-score s_c. When a cutoff score s_c is applied, the portfolio manager accepts all applicants with scores $s \geq s_c$. Following Oliver and Wells [7], we construct the portfolio metrics. Using Bayes Theorem, the portfolio expected revenue is $E_S[R(s_c)] = \int_{s_c}^{\infty}(r_L - c_D)p(G|s)f(s)ds = p_G(r_L - c_D)F^c(s_c|G)$. Similarly, the expected loss for the portfolio may be determined as follows, $E_S[L(s_c)] = \int_{s_c}^{\infty}(l_D + c_D)p(B|s)f(s)ds = p_B(l_D + c_D)F^c(s_c|B)$. It follows that the portfolio expected profit is $E_S[P(s_c)] = E_S[R(s_c)] - E_S[L(s_c)] = p_G(r_L - c_D)F^c(s_c|G) - p_B(l_D + c_D)F^c(s_c|B)$. Given a cutoff score, the expected volume is $E[V(s_c)] = \int_{s_c}^{\infty} 1f(s)ds = F^c(s_c)$. Note, all the portfolio metrics defined thus far such as the expected portfolio revenue, loss, profit, and volume are under the assumption of a single cutoff-score policy.

The first-order derivative of portfolio revenue with respect to portfolio volume is,

$$\frac{\delta E_S[R(s_c)]}{\delta E_S[V(s_c)]} = \frac{(r_L - c_D)p(G|s_c)f(s_c)}{1f(s_c)} = (r_L - c_D)p(G|s_c).$$

Since $(r_L - c_D)p(G|s_c)$ is a non-negative increasing function of s_c and since $E_S[V(s_c)]$ is a decreasing function of s_c, it follows that portfolio revenue is an increasing concave function of portfolio volume. Similarly, portfolio loss is an increasing convex function of portfolio volume.

2.3 Dominance and Efficient Frontier

Given that the portfolio manager is operating under two portfolio metrics, we say an operating point dominates another under the following conditions: (i) if the dominating operating point has one portfolio metric that is higher and the other metric being equal or less, or (ii) if both portfolio metrics are higher for the dominating operating point.

We illustrate the concept of efficient frontier in the expected profit-volume (EPV) space. Suppose we construct the feasible region in the expected profit-volume space. The upper convex hull is called the EPV curve. This is illustrated in Figure 1, the curve extending from $(0, 0)$ to B. Point A on the curve is the maximum expected profit point. However, a portfolio manager with a dual objective of maximizing expected profit and expected volume will operate on the curve $\bar{AB}$. The curve $\bar{AB}$ are the maximal set of operating points that are not dominated by other operating points.

In this paper, we aim to construct the expected profit-expected loss (EPL) curve, which is the upper convex hull of the feasible region on the expected profit-loss space. The efficient

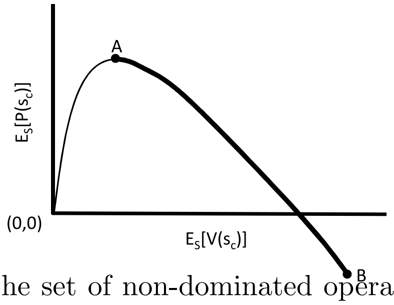


Figure 1: Efficient frontier is the set of non-dominated operating points, $\bar{AB}$ on the convex hull.

frontier on the EPL curve are the maximal set of operating points that are not dominated by other operating points.

3 Decision Models in the EPL Space

In this section, we consider the case of a portfolio manager who is faced with the task of creating a portfolio with objectives in the profit-loss space. The portfolio manager has access to a scorecard that is used to forecast the default risk of applicants, probabilities associated with the occurrence of each economic scenario, and models to evaluate business metrics associated with the portfolio. In this section, we construct the EPL curve and hence the efficient frontier in the EPL space.

Let economic conditions be a random variable W with two mutually exclusive and collectively exhaustive outcomes, economic scenario $W = 1$ and economic scenario $W = 2$. Let q and $1 - q$ be the probability of scenario 1 and 2 occurring, respectively. Given a cutoff score s, let $E_S[R(s)|i]$ and $E_S[V(s)|i]$ be the expected revenue and expected volume conditional under economic scenario i. Therefore, the unconditional expected revenue is $E_W[E_S[R(s)|i]] = qE_S[R(s)|W = 1] + (1 - q)E_S[R(s)|W = 2]$. Similarly, the unconditional expected volume is $E_W[E_S[V(s)|i]] = qE_S[V(s)|W = 1] + (1 - q)E_S[V(s)|W = 2]$. Since expected volume is independent of economic conditions, $E_W[E_S[V(s)|i]] = E_S[V(s)]\ \forall i$[3]. Since $E_S[R(s)|W = i]$ is increasing and concave with respect to $E_S[V(s)|W = i]\ \forall i$, it follows that $E_W[E_S[R(s)|i]]$ is increasing and concave with respect to $E_S[V(s)]$. To summarize,

$$\frac{\partial E_W[E_S[R(s)|i]]}{\partial E_S[V(s)]} > 0 \text{ and } \frac{\partial^2 E_W[E_S[R(s)|i]]}{\partial E_S[V(s)]^2} < 0. \tag{1}$$

Similarly, we can construct the expected loss vs. expected volume curve. Given a cutoff score s, let $E_S[L(s)|i]$ be the expected loss under economic scenario i. Therefore, the unconditional expected loss is $E_W[E_S[L(s)|i]] = qE_S[L(s)|W = 1] + (1 - q)E_S[L(s)|W = 2]$. Since $E_S[L(s)|W = i]$ is increasing and convex with respect to $E_S[V(s)|W = i]\ \forall i$, it follows that $E_W[E_S[L(s)|i]]$ is increasing and convex with respect to $E_S[V(s)]$. To summarize,

$$\frac{\partial E_W[E_S[L(s)|i]]}{\partial E_S[V(s)]} > 0 \text{ and } \frac{\partial^2 E_W[E_S[L(s)|i]]}{\partial E_S[V(s)]^2} > 0. \tag{2}$$

[3] The volume is determined at portfolio creation stage when accounts are booked, which is prior the occurrence of the future economic condition.

It follows from (1) and (2),

$$\begin{aligned}
\frac{\partial^2 E_W[E_S[P(s)|i]]}{\partial E_S[V(s)]^2} &= \frac{\partial}{\partial E_S[V(s)]}\frac{\partial E_W[E_S[P(s)|i]]}{\partial E_S[V(s)]} \\
&= \frac{\partial}{\partial E_S[V(s)]}\left(\frac{\partial(E_W[E_S[R(s)|i]] - E_W[E_S[L(s)|i]])}{\partial E_S[V(s)]}\right) \\
&= \frac{\partial^2 E_W[E_S[R(s)|i]]}{\partial E_S[V(s)]^2} - \frac{\partial^2 E_W[E_S[L(s)|i]]}{\partial E_S[V(s)]^2} < 0.
\end{aligned}$$

Therefore, the EPV curve is concave.

From the second order derivative in expression (2), it follows that,

$$\frac{\partial^2 E_S[V(s)]}{\partial E_W[E_S[L(s)|i]]^2} < 0. \tag{3}$$

Our interest in this paper is to construct the non-dominated set of operating points on the EPL curve.

Proposition 1 *The set of non-dominated operating points on the EPL curve is concave and increasing.*

Applying the chain-rule, we determine the second order derivative of this curve,

$$\begin{aligned}
&\frac{\partial^2 E_W[E_S[P(s)|i]]}{\partial E_W[E_S[L(s)|i]]^2} = \frac{\partial}{\partial E_W[E_S[L(s)|i]]}\left(\frac{\partial E_W[E_S[P(s)|i]]}{\partial E_S[V(s)]}\frac{\partial E_S[V(s)]}{\partial E_W[E_S[L(s)|i]]}\right) \\
&= \frac{\partial E_W[E_S[P(s)|i]]}{\partial E_S[V(s)]}\frac{\partial^2 E_S[V(s)]}{\partial E_W[E_S[L(s)|i]]^2} + \left(\frac{\partial E_S[V(s)]}{\partial E_W[E_S[L(s)|i]]}\right)^2 \frac{\partial^2 E_W[E_S[P(s)|i]]}{\partial E_S[V(s)]^2}.
\end{aligned} \tag{4}$$

In addition, the first order derivative is

$$\frac{\partial E_W[E_S[P(s)|i]]}{\partial E_W[E_S[L(s)|i]]} = \frac{\partial E_W[E_S[P(s)|i]]}{\partial E_W[E_S[V(s)]]}\frac{\partial E_W[E_S[V(s)]]}{\partial E_W[E_S[L(s)|i]]}.$$

It follows that

$$\frac{\partial^2 E_W[E_S[P(s)|i]]}{\partial E_W[E_S[L(s)|i]]^2} < 0 \text{ and } \frac{\partial E_W[E_S[P(s)|i]]}{\partial E_W[E_S[L(s)|i]]} > 0 \text{ wherever } \frac{\partial E_W[E_S[P(s)|i]]}{\partial E_W[E_S[V(s)]]} > 0.$$

Furthermore, as $s_c \to \infty$ then $E_W[E_S[P(s)|i]] = 0$ and $E_W[E_S[L(s)|i]] = 0$. As s_c decreases both $E_W[E_S[P(s)|i]]$ and $E_W[E_S[L(s)|i]]$ increase until the maximum profit point is reached. After which, expected profit does not increase in expected loss [4]. The EPL curve from (0,0) to the maximum profit point is increasing and concave and therefore, this curve is the maximal set of operating points that are not dominated by other operating points under the single cutoff-score strategy. ■

[4] The EPV curve under a single cutoff-score results in a single maximum profit point [7]. Therefore, the expected profit, $E_W[E_S[P(s)|i]]$ decreases with further decrease in cutoff-score on the EPL curve.

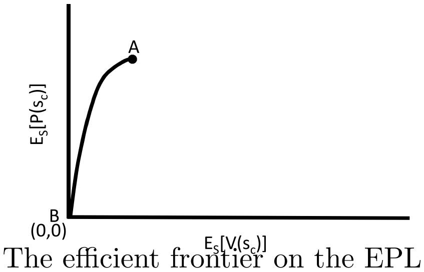


Figure 2: The efficient frontier on the EPL space $\bar{BA}$.

By Proposition 1, the EPL curve is concave and increasing in the maximal set of non-dominated operating points under the single cutoff-score strategy. Suppose no other strategy dominates the single cutoff-score strategy in the EPL space, then the maximal set of operating points with the single-cutoff score strategy is the efficient frontier of the EPL space.

Proposition 2 *Given $s_1 > s_2$, the portfolio manager operating under the profit-loss objective pair, would prefer to accept an applicant with score s_1 over one with score s_2.*

Since $s_1 > s_2$, it follows that $p(G|s_1) > p(G|s_2)$ and $p(B|s_1) < p(B|s_2)$. Therefore,

$$\frac{(r_L - c_D)p(G|s_1) - (l_D + c_D)p(B|s_1)}{(l_D + c_D)p(B|s_1)} > \frac{(r_L - c_D)p(G|s_2) - (l_D + c_D)p(B|s_2)}{(l_D + c_D)p(B|s_2)}.$$

Hence, given two applicants, one with score s_1 and the other with score s_2, a portfolio manager would prefer the applicant with score s_1 over s_2. ■

Proposition 2 implies a single cutoff-score strategy would result in the maximum expected portfolio profit for a given expected portfolio loss. Therefore, the set of non-dominated operating points under the single cutoff-score strategy is the efficient frontier. The efficient frontier is illustrated in Figure 2. The efficient frontier is the curve $\bar{BA}$.

4 Summary

In this paper, we extend the literature in constructing an efficient portfolio in the expected profit-expected loss space. While market share remains an important objective for any financial institution, recent concerns about cash flow management in financial institution has brought more focus in to loss as a primary objective. Oliver and Wells [7] showed methods to construct the efficient frontier in the EPL space under the assumption of a single economic conditions. In this paper, we show methods to construct the efficient frontier in the EPL space under the multiple economic condition assumption, and show all efficient points are constructed using the single cutoff-score strategy. There are some extension to this work that is not illustrated here. A particular important extension is the case where there are multiple scorecards with each one built specifically for different economic condition. Another extension is to incorporate variance on the profit and variance in the loss in the decision making process.

Bibliography

[1] Beling, P., Covaliu, Z., and Oliver, R., “Optimal Scoring Policies and Efficient Frontiers.”, *Journal of the Operational Research Society*, **56**, pp. 1016–1029, 2005.

[2] Capon, N., “Credit Scoring Systems: a Critical Analysis.”, *Journal of Marketing*, **46**, pp. 82–91, 1982.

[3] De Andrade, F.W.M. and Silva, R.G., “Use of Macro-Economic Factors in Credit Scoring - Application to Point-in-time Risk Evaluation of SMEs.”, *Proceedings of Credit Scoring and Credit Control Conference X*, Edinburgh, 2007.

[4] Hand, D.J. and Henley, W.E., “Statistical Classification Methods in Consumer Credit Scoring”, *Journal Royal Statistical Society Series A*, **160**, pp. 523-541, 1997.

[5] Hoadley, B. and Oliver R.M., “Business Measures of Scorecard Benefit.”, *IMA Journal of Mathematics Applied to Business and Industry*, **9**, pp. 55–64, 1998.

[6] Lewis, E.M., *An Introduction to Credit Scoring*, Fair Isaac and Co., San Rafael, California, 1992.

[7] Oliver, R. and Wells, E., “Efficient Frontier Cutoff Policies in Credit Portfolios.”, *Journal of the Operational Research Society*, **52**, pp. 1025–1033, 2001.

[8] Overstreet G.A., Bradley, E.L., and Kemp R.S., “The Flat-maximum Effect and Generic Linear Scoring Models: a Test.”, *IMA Journal of Mathematics Applied in Business and Industry*, **4**, pp.97–109, 1992.

[9] Rajaratnam, K., Beling, P., and Overstreet, G., “Scoring Decisions in the Context of Economic Uncertainty.”, *Journal of the Operational Research Society*, **61**, pp. 421–429, 2010.

[10] Rajaratnam, K., Beling, P., and Overstreet, G., “Scoring Decisions under Regulatory Capital Constraints.”, *Proceedings of Credit Scoring and Credit Control Conference XII*, Edinburgh, 2011.

[11] Thomas, L.C., Edelman, D., and Crook, J.N., “Credit Scoring and Its Applications.”, *Society for Industrial and Applied Mathematics*, Philadelphia, USA, 2002.

[12] Zandi, M., “Incorporating Economic Information into Credit Risk Underwriting”, in: Mays, E., Editor, *Credit Risk Modeling*, Glenlake Publishing, Chicago, pp. 155–168, 1998.

Proceedings of the 2012 ORSSA Annual Conference
pp. 48 – 57
www.orssa.org.za/wiki/uploads/Conf/2012ORSSAConferenceProceedings.pdf

ORSSA Proceedings
ISBN
978-07972-1406-4
© 2012

Finding the best pass-receiving position in the RoboCup Small-Size League

M Yoon[1] T Lane-Visser[2]

Abstract

The RoboCup, as an attempt to provide a common platform for various research fields, offers numerous problems for decision-making and optimisation. Specifically, the problem of finding a good position for a pass-receiving robot is dealt with in this paper. Besides the challenge of developing suitable evaluation criteria to assess various field positions, another difficulty for approaches to solving this problem is the time constraint. The solution should be found quickly enough to cope with a live field situation that is continuously and rapidly changing.

In this paper a set of criteria to evaluate each position on the field, and a Cuckoo Search (CS) metaheuristic model are proposed to find the best field position for a pass-receiving robot. Further, the run length of the algorithm is balanced between the quality of the solution and the agility in finding a solution. The solutions found by the CS algorithm were compared to explicit enumeration results for three game situations. The comparison proved that the proposed CS algorithm can find sufficiently good pass-receiving locations for RoboCup Small-Size games.

Key words: RoboCup, metaheuristic, Cuckoo Search, pass-receiving strategy

1 Introduction

The development of a team of fully autonomous soccer-playing robots (the RoboCup [1]) provides an interesting test-bed for the application of various optimisation techniques, as well as many other topics in Robotics and Artificial Intelligence (AI). The RoboCup is divided into five different leagues; this research is focused on the RoboCup Small-Size League (SSL). In the RoboCup SSL, teams consisting of five cylinder-looking robots play soccer games using an orange golf ball on a pitch of 6.05 m × 4.05 m. Two video cameras mounted above the field are used for world perception, and a decision-making module, run on an off-field computer, produces team strategies for the robots and sends commands to the robots. Many optimisation methods can be used to help this module make decisions, as there are a large number of decisions that need to be optimised.

[1] Corresponding author: Department of Industrial Engineering, Stellenbosch University, Private Bag X1, Matieland, 7602, South Africa, email: 17090792@sun.ac.za
[2] Department of Industrial Engineering, Stellenbosch University, Private Bag X1, Matieland, 7602, South Africa, email: tanyav@sun.ac.za

The most important decision in a soccer game is probably the one where the player who has the ball must decide whether to shoot or to pass the ball. The decision of field location for a teammate who doesn't have the ball, however, is also critical, as Kyrylov *et al.* [2] stated. Because the defending team would make every effort to obstruct the goal from the player who has the ball, the decision of a teammate attacking robot to locate itself in a promising field position in order to "receive-a-pass-and-shoot-immediately" becomes essential for the team to win a game. This paper proposes a method to find such a field position, which can be employed by the decision-making module of a RoboCup SSL team.

This article is structured as follows: various methods to solve this problem as found in the literature, as well as a discussion of the approach used in this paper, are introduced in Section 2. Section 3 presents a set of criteria to evaluate each position on the field, with a few examples of entire field explicit enumeration based on these criteria. Section 4 proposes a metaheuristic model to solve the problem in a shorter amount of time than through the enumeration, followed by a comparison of the results between the metaheuristic and the explicit enumeration. Section 5 concludes the paper.

2 Problem domain: finding the best pass-receiving position

The earliest approach to finding the best position for a pass is to use a set of logical "IF-THEN-ELSE" rules to determine which teammate to pass to. Later, the pass strategies were advanced to find a good position on the field, not necessarily a teammate, to pass to. That is, the passer kicks the ball to a strategic point on the field, and one of the teammates moves to this position to receive the pass. In this approach, a set of criteria is required to evaluate each position on the field. CMDragons, one of the most successful teams in the RoboCup SSL, used a pass position evaluation function, defined over the field [3]. They evaluated the entire soccer field by dividing it into approximately 2400 grids to find the best position for a pass. This method, however, is only applicable when the associated calculation time is acceptable. The field size here was relatively small (4.9 m × 3.8 m) when the algorithm was developed. Field size has, however, occasionally been increased since then to keep the teams challenged.

In another league, the Simulation League, where two teams of eleven autonomous agents play soccer in a virtual soccer stadium of 105 m × 70 m [4], it is impossible to search the entire field within an appropriate amount of time. Teams in the Simulation League are forced to search within a restricted decision space, which might not include all good solutions. To mitigate this problem, Kyrylov *et al.* [2] proposed an algorithm to carefully generate reasonably small number of prospective alternatives. Using an optimisation method and multi-criteria decision making (MCDM) theory, the generated alternatives are searched for non-dominated solutions (Pareto optimal solutions), out of which one alternative is ultimately selected.

Another approach, in which an optimisation method was used to solve this problem, is seen in [5]. In their Strategic Positioning with Attraction and Repulsion (SPAR) algorithm, Veloso *et al.* took into account the distance to the position of teammate robots, opponent robots, the ball, and the opponent goal as four different objectives to be maximised or minimised. They combined them into a single objective function by aggregating them with various weights, and solved the optimisation problem to find a good strategic position for a pass-receiving player. However, the concept of an "open-angle", an obstruction-free angle toward the opponent goal from the point in question, was not considered as one of the criteria. The authors believe that this open-angle plays an important role in the pass-receiving robot's ability to win a goal by shooting the ball as soon as it is received and, therefore, that the concept should be included in such an optimisation model.

3 Criteria for a good pass-receiving position

To reduce the infinite decision space to a manageable size, the field was divided into 605 × 405 grids, each 10 mm × 10 mm, and the quality of the field position was evaluated at each node of the grid. In other words, it was evaluated at every 10 mm in both X and Y directions. The decision variables are thus the pair *(x, y)* in the X-Y coordinates system shown in Figure 1. The centre of the field is the origin of the coordinates. The top left corner of the field is (-3025, 2025) and the bottom right corner is (3025, -2025) in this coordinate system.

The fitness of a specific field position is a function of the performance against a set of decision criteria and it indicates how desirable this location is. Constraints should also be considered to exclude some of the positions in the field where the robot should not be located.

3.1 Fitness

Four criteria were considered to calculate the fitness of each position on the field as shown in Equation 1. They are multiplied by corresponding weights, and are then aggregated to form a single fitness function to be maximised. Note that criteria 3 and 4, the distance from the position to the ball and to the opponent goal, respectively, are subtracted from the fitness value because the shorter they are, the better the position is. When the fitness is maximised, these negative terms will be minimised. The weights are to be adjusted by the human decision-maker according to the priority ranking of the criteria. In this study, w_1=1, w_2=w_3=w_4=0.001 was used in order to scale the function elements.

$$\text{Fitness of position P} = \text{w}_1 * \text{OpenAngle} + \text{w}_2 * \sum_{\text{i}=1}^{4} \overline{\text{PO}_\text{ı}} - \text{w}_3 * \overline{\text{PB}} - \text{w}_4 * \overline{\text{PG}} \qquad \text{[Eq. 1]}$$

- OpenAngle : the unobstructed angle toward the opponent goal from position P (in degrees)
- $\overline{PO_\iota}$: the Euclidian distance between position P and opponent player i (in millimetres) (i=1,…,4)
- $\overline{PB}$: the Euclidian distance between position P and the ball (in millimetres)
- $\overline{PG}$: the Euclidian distance between position P and the opponent goal (in millimetres)
- w_j: weight to be adjusted for the j^{th} decision criterion (j=1,..,4)

3.2 Constraints

There are some positions on the field that the robot should not consider moving to. Six constraints were applied to help avoid those positions. They are listed below, with illustrations provided in Figure 2. In Figure 2 we consider the game situation where the yellow robot (3) has the ball and we are determining the optimal pass receiving location for the yellow robot (2). They are attacking the right hand goal.

1. The position must not be already occupied by other players. (See the yellow circle in Figure 2)
2. The position should not be so far from the current position of the pass-receiving robot that the robot cannot arrive in time to receive the ball. (See the red circle in Figure 2)
3. The pass-receiving robot should be able to reach this point faster than any opponent robot. (See the blue line in Figure 2)
4. Other players should not obstruct the path between the ball and the pass-receiving player. (See the green parabola in Figure 2)
5. The reflection angle between the pass line and the shoot line should be acute, so that an immediate shot at goal after receiving the pass is relatively easy. (See the purple circle and lines in Figure 2. The purple lines show the pass line and the shoot line.)

6. The position must not block the goal shoot angle of the teammate robot that currently has the ball. (See the red dotted lines in Figure 2)

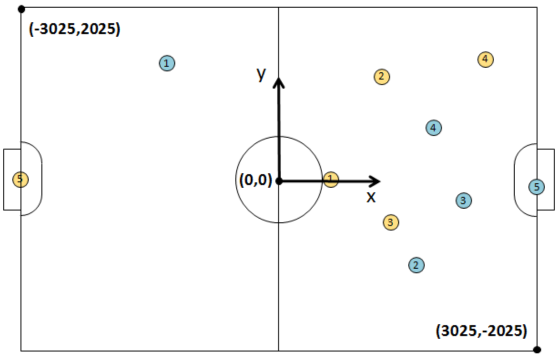


Figure 1: *Example of a field situation*

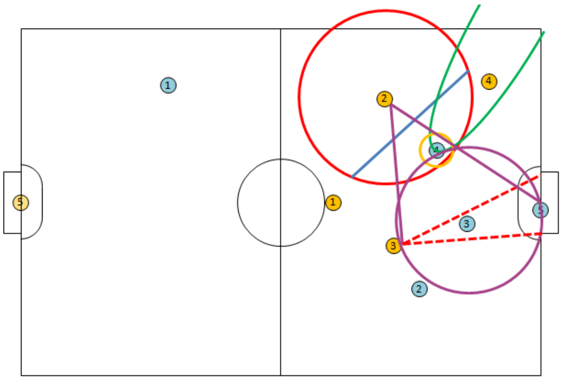


Figure 2: *Constraints applied*

These constraints were implemented by a set of circles, a cone, a parabola and lines in the *X-Y* coordinate plain as seen in Figure 2. Equations 2 to 7 represent examples of the formulas implemented for each constraint, in order. Particularly, constraint 2 restricts the decision space to the inside of a circle centred at the current position of the pass-receiving robot. The radius of the circle, as with other constraints, is up to the decision-maker. In this research, the radius of the circle was set to 1000 mm.

$$(x-1800)^2+(y-600)^2 \geq 190^2 \qquad \text{[Eq. 2]}$$

$$(x-1200)^2+(y-1200)^2 \leq 1000^2 \qquad \text{[Eq. 3]}$$

$$(x-1200)^2+(y-1200)^2 \leq (x-1800)^2+(y-600)^2 \qquad \text{[Eq. 4]}$$

$$(-(x-1800)*\sin 65.56+(y-600)*\cos 65.56)^2 \geq 4*10*((x-1800)*\cos 65.56+(y-600)*\sin 65.56) \qquad \text{[Eq. 5]}$$

$$(x-2150)^2+(y-(-250))^2 \geq 785000 \qquad \text{[Eq. 6]}$$

$$(y-0.5*(x-1300)+500)*(y-0.88235*(x-1300)+500)) \geq 0 \qquad \text{[Eq. 7]}$$

3.3 Field evaluation examples

A number of potential game situations were supposed and implemented to test the model formulation described above. Due to the page restraint, however, only three game situations are presented in this paper to show examples of how the field was evaluated. All cases use the same location distribution of robots depicted in Figure 1, where the yellow circles represent the attacking team's robots, while the blue circles stand for the defending opponents. It is always assumed that one of the attacking teammates has possession of the ball, and that they are attacking the goal on the right. In the first game situation (Figure 3), robot 2 is assumed to have possession of the ball, and the robot for which we are determining the best pass-receiving position is robot 3, while in the second and third game situations (Figures 4 and 5), robot 3 is supposed to have the ball, and robot 1 and robot 2 are considered as the pass-receiving robot in each case. The criteria described in 3.1 and 3.2 were used in the field evaluation. If the field position in question violated one of the constraints, the fitness was set to zero for that position. The results are presented in Figure 3 to Figure 5, where the ball is represented as a white square, and the pass-receiving robot is shown as a white circle. The colour bar at the right side of the figures represents the fitness scale used, where a redder colour indicates a better fitness value. Most parts of the field are dark blue, which means the fitness at that location is equal to zero, due to violation of one or more of the constraints.

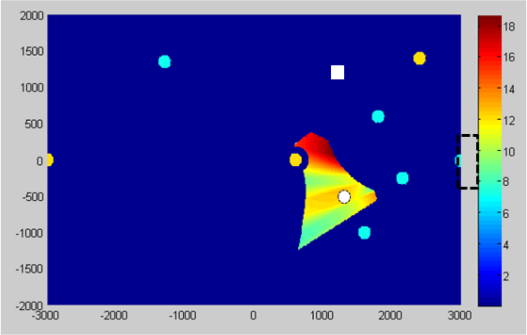


Figure 3: *The evaluated field in the first game situation*

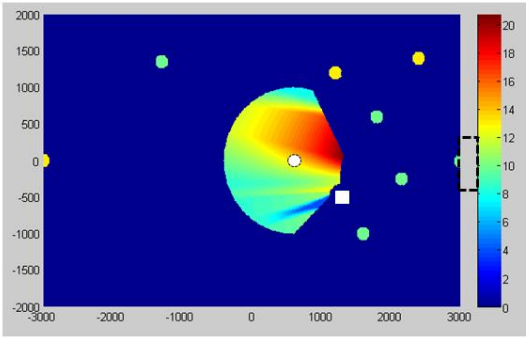


Figure 4: *The evaluated field in the second game situation*

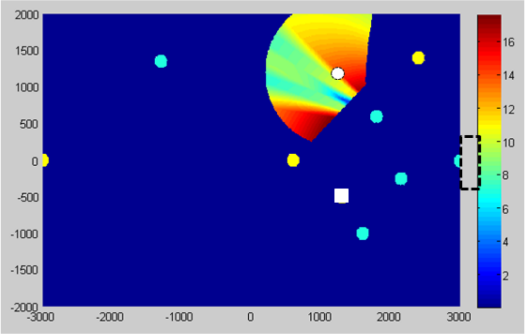
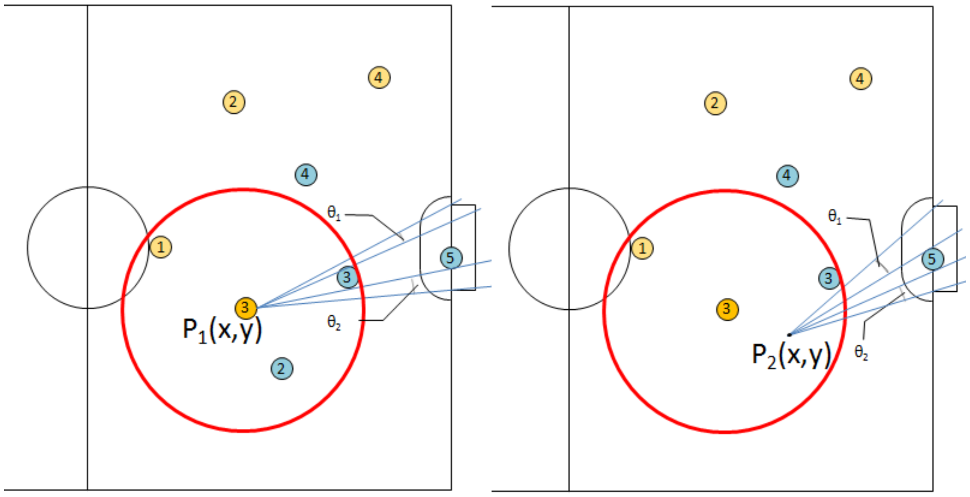


Figure 5: *The evaluated field in the third game situation*

4 Developing the metaheuristic model

In section 3, it was demonstrated that the best pass-receiving position for a certain robot in a given situation can be computed by explicit enumeration. The problem with this approach is the time required to compute all possible field location evaluations. It takes more than 30 seconds, which is unacceptable considering the highly dynamic nature of RoboCup Small-Size League games. An intelligent optimisation model is required to find the optimal solution (or a near-optimal solution at least) in a shorter amount of time. A nonlinear model would be ideal for this purpose and was attempted. It turned out, however, that it is extremely complicated to formulate the open-angle as a generic function of the decision variables *(x, y)*. Even though the positions of all robots and the ball are fixed and, accordingly, they are defined as constants in the model, the way to measure the open-angle varies based on the active position of interest, as shown in Figure 6. Defence robot 3 should be considered in the calculation of the open-angle from the point P_1 in Figure 6(a), while only the defensive robot 5 affects to the open-angle from the point P_2 in Figure 6(b).

Metaheuristics are a good alternative in this case because, in metaheuristic models, a candidate solution is chosen and the fitness is calculated for the selected solution. The open-angle criterion in a metaheuristic model is not to be expressed as a function of decision variables *(x, y)*, but it is simply calculated when a point *(x, y)* is chosen.



(a) (b)

Figure 6: *The difference in measurement of the open-angle for different points of interest, (a) with position of interest at point P_1 and (b) at point P_2.*

4.1 The Cuckoo Search Algorithm

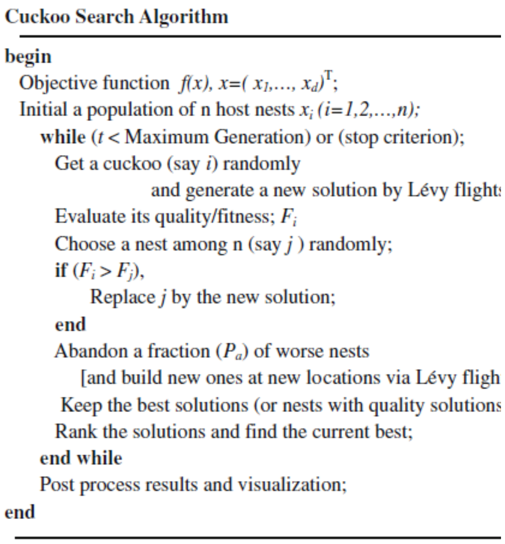
The Cuckoo Search (CS), proposed by Yang *et al.* [6, 7], is a nature-inspired metaheuristic algorithm. It is reported to be very simple and powerful, superior to Genetic Algorithms (GA) and Particle Swarm Optimisation (PSO) for multimodal objective functions [6]. In nature, cuckoos lay their eggs in other birds' nests. If the host bird discovers the cuckoo egg, they throw out the alien egg or abandon the nest and build a new nest. In the Cuckoo Search Algorithm, each egg in a nest represents a solution. A cuckoo egg stands for a new solution. If the cuckoo egg laid in a host nest is superior to the host egg, it replaces the egg. The replaced cuckoo eggs are found by the host bird with a probability of P_a. This discovery is implemented in the algorithm as a kind of elitism. A fraction of P_a worse nests are abandoned, and new solutions are developed. In the CS algorithm, each new solution is developed by means of Lévy flights. A Lévy flight is a random walk in which the step-lengths have a probability distribution that is heavy-tailed. Thus, a new solution is a random walk from the current solution in the space of possible solutions. The direction is chosen randomly and the step-size is chosen from the Lévy distribution. The heavy-tail in the distribution pushes the algorithm to explore the space, while the discovery and abandonment enhance the degree of exploitation.

The pseudo code for the Cuckoo Search presented in [7] is shown below in Figure 7. In this study, the algorithm was modified for better efficiency, in the way that the authors implemented it in their released code [8]. Here, in each generation, *n* new cuckoo eggs are generated (*n* is the number of host nests) instead of having only one cuckoo egg to find a host nest randomly, as seen in Figure 7. The Cuckoo Search in this form seems quite similar to the (`, λ) evolution strategy (shown in Algorithm 18 in [9]) when `=λ=the number of host nests. The difference appears, however, in the fact that the parent (the host egg) competes with its own child (the cuckoo egg) to survive in the next generation. Also, the existence of another elitism, given by the discovery and the abandonment of the egg, differentiates it from the simple (`, λ) evolution strategy.

The fitness function and constraints described in section 3 were used during implementation of the CS algorithm. The fitness was set to zero if the position did not satisfy one the constraints, except for constraint 2. For the second constraint, the algorithm was implemented so that it searches only those positions that satisfy it. When a new solution is generated, if the step-size chosen from the Lévy distribution is so large that it violates constraint 2, the step-size is shortened, so that the new solution is on the verge of the constraint. The number of host nests, n=25, and the probability of the abandonment, P_a=0.25, were used in this study, as recommended in [6].

4.2 Run length of the algorithm

Ideally, the algorithm should use as small a number of iterations to converge as possible. The number of iterations represents a balance between the quality of the solution and the calculation time required. If the algorithm is run long enough, it will be able to find the optimal solution. On the other hand, in order for the algorithm to be effectively used by the decision-making module for a RoboCup SSL team, the authors believe that the calculation time should be less than one second, at most. To determine the preferred number of iterations, the developed CS algorithm was run for the first game situation for an arbitrarily long amount of time. This was repeated 100 times. The average fitness of the best position over the 100 runs at each iteration count is shown in Figure 8, which shows that almost 2000 iterations are required to reach convergence. However, the quality of the solution could be sacrificed to quicken results for the purpose of the RoboCup SSL. Solutions which achieve a fitness of more than 95% of the fitness of the optimal solution are found within 500 iterations. After some additional experiments with the other cases, the appropriate length for a run of the algorithm was determined to be 800 iterations.



```
Cuckoo Search Algorithm
begin
  Objective function $f(x)$, $x=(x_1,\dots,x_d)^T$;
  Initial a population of n host nests $x_i$ $(i=1,2,\dots,n)$;
    while ($t$ < Maximum Generation) or (stop criterion);
      Get a cuckoo (say $i$) randomly
                and generate a new solution by Lévy flight:
      Evaluate its quality/fitness; $F_i$
      Choose a nest among n (say $j$ ) randomly;
      if ($F_i > F_j$),
          Replace $j$ by the new solution;
      end
      Abandon a fraction ($P_a$) of worse nests
          [and build new ones at new locations via Lévy fligh
       Keep the best solutions (or nests with quality solutions
      Rank the solutions and find the current best;
    end while
    Post process results and visualization;
end
```

Figure 7: *Pseudo code of the Cuckoo Search [7]*

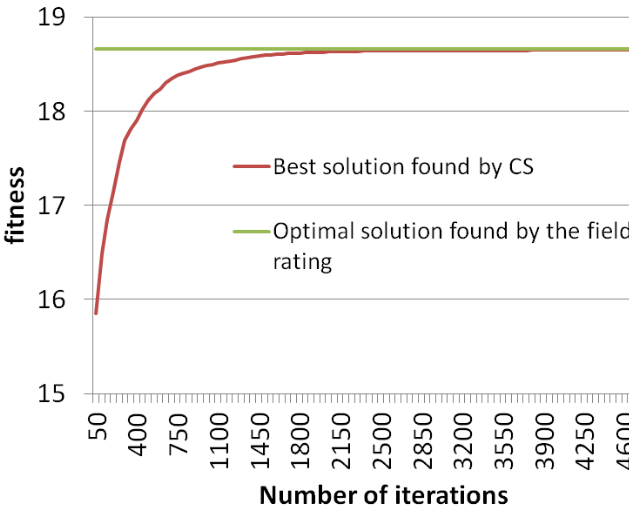


Figure 8: *Convergence of the Cuckoo Search*

4.3 Result from Cuckoo Search

The CS algorithm was applied for each of the three game situations mentioned before. The algorithm was run 100 times for each game situation, and 800 iterations were executed for each run. Table 1 summarises the results along with the best position (optimal solution) found by the enumeration in section 3.3. The average, maximum and minimum fitness found in the 100 runs of the CS algorithm are presented, as well as some other useful information (such as the number of solutions better than 95% of the fitness of the optimal solution, the number of times that the true optimal solution was found out of the 100 runs, and the average calculation time).

Table 1: *Results of the Cuckoo Search*

	Game situation 1	Game situation 2	Game situation 3
Optimal Solution found by the explicit enumeration	18.6604 at (1150, 80)	20.7560 at (1320, 60)	17.6127 at (1190, 590)
Average of CS	18.3983 at (1121, 98)	20.5028 at (1295, 62)	17.3406 at (1091, 516)
Maximum of CS	18.6604 at (1150, 80)	20.7560 at (1320, 60)	17.6127 at (1190, 590)
Minimum of CS	17.2212 at (950, 120)	19.7251 at (1220, 110)	16.4666 at (990, 590)
# of solutions better than 95% of optimal fitness	98 out of 100	100 out of 100	98 out of 100
# of times the exact optimal solution was found	3 out of 100	8 out of 100	2 out of 100
Average CPU time	0.85 seconds	0.91 seconds	0.70 seconds

In all game situations, the optimal solution was found only a few times out of the 100 runs of the algorithm. However, the average fitness shows little difference from the fitness of the optimal solution, and 98 times or more out of the 100 runs, the algorithm was able to find solutions better than 95% of the fitness of the optimal solution. More importantly, even though the optimal solution was not found, the best solution from each run was “sufficiently” near the optimal solution. Here, “sufficiently” means that the best solution found from the Cuckoo Search points to the direction of the optimal solution from the current position of the pass-receiving robot. In other words, in the sense of hill-climbing, even though we don’t get to the top of the highest hill, the best solution found by the CS algorithm was on the highest hill. In the third game situation (shown in Figure 5), for example, we can see there are two high hills around (1200, 600) and (1700, 1000). All the best solutions found in the 100 runs are on the highest hill, around (1200, 600) of which the hill-top is the optimal solution, as shown in Figure 9 (the best solutions found in the 100 runs are represented as blue dots).

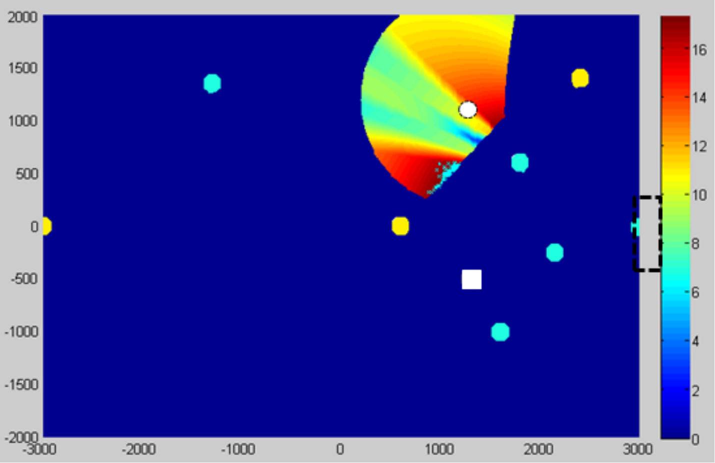


Figure 9: *The distribution of the best solutions found by CS*

Also, we can see that the algorithm finds the solution within one second at most, as desired, when using a single Intelff Core™ i5 CPU M520 at 2.4GHz with 4GB RAM. Further, it is proved that the algorithm reaches convergence faster, without substantial deterioration in the quality of the solution, when the field is divided into slightly bigger grids (402 × 270 grids, each 15 mm × 15 mm), than those used in this paper. This enables the algorithm to carry out the same performance in a shorter amount of time. When applied to the three game situations considered above, an average of 0.15 seconds, 0.03 seconds and 0.11 seconds in each game situation, respectively, could be saved with this approach.

The results analysis from the CS algorithm leads us to conclude that the CS algorithm can be used to determine the best position for a pass-receiving robot in RoboCup SSL games. The appropriate grid size with regards to the accuracy of robots' movement as well as the size of the ball and accuracy of the kicking device promises to be an interesting topic for future research.

5 Conclusions

The decision-making module of a RoboCup SSL team is required to make decisions quickly in reaction to dynamic changes in the game. In this research, the authors have developed an algorithm for the decision-making module to employ when it needs to determine the target position for a robot who is expecting a pass. To find a good position for the robot to receive a pass at, a set of criteria to evaluate each position on the field was firstly proposed. The entire field was then evaluated (using the proposed criteria) by explicit enumeration for three potential game situations. This found the best position, but failed to find it quickly enough. A metaheuristic model based on the Cuckoo Search algorithm was developed in order to find the solution in an acceptable amount of time. Its performance was compared to that of the explicit enumeration and it was shown that the developed algorithm could find sufficiently good solutions in less than a second, which is much quicker than the enumeration could achieve. This means that it can be considered for inclusion in the decision-making module of a RoboCup SSL team.

Bibliography

[1] Kitano, H., Asada, M., Kuniyoshi, Y., Noda, I. and Osawa, E. (1997). RoboCup: the robot world cup initiative. In *Proceedings of the First International Conference on Autonomous Agents (Agents'97)*, New York, 340-347. (ACM Press, New York)

[2] Kyrylov, V., Bergman, D.S. and Greber, M. (2005). Multi-criteria optimization of ball passing in simulated soccer. *Journal of Multi-Criteria Decision Analysis,* 13(2-3): 103-113.

[3] Bruce, J., Zickler, S., Licitra, M. and Veloso, M. (2008). Cmdragons: Dynamic Passing and Strategy on a Champion Robot Soccer Team. In *Proceedings of IEEE International Conference on Robotics and Automation,* 4074-4079.

[4] Soccer Simulation League. Online. Available: http://wiki.robocup.org/wiki/Soccer˙Simulation˙League [Cited June 13th, 2012]

[5] Veloso, M., Stone, P. and Bowling, M. (1999). Anticipation as a key for collaboration in a team of agents: A case study in robotic soccer. In *Proceedings of SPIE Sensor Fusion and Decentralized Control in Robotic Systems II*, 3839.

[6] Yang, X.S. and Deb, S. (2009). Cuckoo search via Lḗvy flights. In *Proceedings of World Congress on Nature & Biologically Inspired Computing (NaBIC 2009)*, India, 210-214. (IEEE Publications, USA)

[7] Yang, X.S. and Deb, S. (2010). Engineering optimisation by cuckoo search. *International Journal of Mathematical Modelling and Numerical Optimisation,* 1(4): 330-343.

[8] Yang, X.S. (2010). Cuckoo Search (CS) Algorithm. Online. Available: http://www.mathworks.com/matlabcentral/fileexchange/29809-cuckoo-search-cs-algorithm [Cited June 18th, 2012].

[9] Luke, S. (2010). *Essentials of Metaheuristics*. Department of Computer Science, George Mason University, USA, 31-32.

Proceedings of the 2012 ORSSA Annual Conference
pp. 58–66
www.orssa.org.za/wiki/uploads/Conf/2012ORSSAConferenceProceedings.pdf

ORSSA Proceedings
ISBN
978-07972-1406-4
© 2012

An implementable routing solution for home-based care in South Africa

NM Viljoen[1]

Abstract

Home-based care (HBC) is an effective service model to reduce the burden on a country's health and welfare systems. In South Africa, the orphaned and vulnerable children crisis has become the focus of HBC programmes. These programmes are mostly within semi-urban settlements where the need is greatest. Successful software solution approaches developed to support HBC routing in other countries are difficult to implement in this low-tech, low-resource environment. A solution approach based on the spacefilling curve heuristic is presented as an easily implementable, adequately performing alternative to improve the routing of daily home visits.

Key words: travelling salesman problem, home-based care, spacefilling curve heuristic, humanitarian operations research

1 Introduction

Home-based care (HBC) has emerged as an effective service model to reduce the burden on public health care and welfare systems in South Africa [9]. The HIV/Aids pandemic has left countless orphaned and vulnerable children (OVC) in its wake, making child-headed households a common occurrence [13]. Studies have shown that caring for these OVC within the context of their extended families and communities is much more conducive to their quality of life than institutionalised care [16]. In response, many non-governmental organisations (NGOs), community-based organisation (CBOs) and faith-based organisations (FBOs) are providing home-based care to OVC throughout the country. Regular visits are made to OVC households by careworkers to offer material and emotional support and various other services for which a child may need an adult's help (such as applying for an ID document or grants) [8]. Other HBC programmes particularly relieve the burden on public health care through palliative care for terminally ill patients, assisting people during convalescence or those living with HIV/Aids or other debilitating diseases, supporting people with mental illness and those with mental or physical disabilities, and assisting the elderly [13].

For HBC to be a viable alternative to hospitalisation or institutionalisation it must maintain consistent and adequate levels of service delivery. NGOs, CBOs and FBOs have tight resource

[1] Corresponding author: CSIR Built Environment, South Africa, PO Box 395, Pretoria 0001, email: nviljoen@csir.co.za

constraints and any additional funding or time is typically applied to serve immediate beneficiary needs [3, 8]. Little resources are left to devote to managerial tasks or technology solutions. In addition the staff and volunteers within these organisations typically have skill sets focussed on the beneficiary needs instead of skill sets focussed on technology solutions, business management etc. This means that these organisations often lack the helpful technology and support services that for-profit companies use to improve cost efficiency and service delivery.

The routing and scheduling of home visits is a pertinent headache for HBC organisations [8]. Routing and scheduling problems have been studied for decades and today there are many software packages (both commercial and open source[2]) that can solve generic routing-type problems. And if generic solutions do not suffice, a number of heuristic solution methods have been developed to address the specific context of HBC and these can be used to create customised solutions [1, 5, 6, 7, 8, 9, 10]. Notwithstanding, most not-for-profit organisations do not make use of such technology, probably because the commercial packages are too expensive (and daunting) and the open source packages or customised algorithms require significant computer programming and/or engineering expertise. In South Africa, most HBC programmes serve poorer communities within informal settlements where the need is greatest. Careworkers are sourced from these communities and usually do not have private transport, smartphones or computers with internet connectivity, so even the recent developments in the functionality of web-based services such as GoogleMaps and GoogleEarth are not generally accessible to careworkers.

Rahman and Smith [15] comment that traditional OR methodology is not necessarily suitable to problems in developing countries as an overemphasis on OR techniques as opposed to OR thinking results in solutions that do not adequately account for culture and context. They maintain that for OR technology to be successfully implemented in developing countries it should be: appropriate to the local users; applicable to the local culture; and relevant to the local problem. Two imperatives in developing such technology is the involvement of local analysts throughout development and effective communication with local decision makers. They quote Woolsey [19, 15:267] who stated: "People would rather live with a problem they cannot solve than accept a solution they cannot understand."

Strebel [16], in her study of numerous community based OVC initiatives in Sub-Saharan countries, found that it is imperative for external organisations to proactively drive the empowerment of and hand-over of initiatives to the community, enabling the replication of best practise models and the sustainability of OVC care programmes. This observation stresses that any technology developed to facilitate HBC services should be relevant and applicable to the community itself, and not just the initiating organisation.

This paper discusses the appropriateness of the spacefilling curve heuristic to the routing of home visits in the South African HBC environment. Section 2 elaborates on the HBC environment based on a case study of a specific OVC care programme. Section 3 briefly discusses some recent solution approaches developed to address routing and scheduling within various HBC environments. Section 4 presents the spacefilling curve heuristic. Section 5 concludes the paper by discussing the appropriateness of the space filling curve heuristic and suggesting a way forward in terms of implementation.

2 The South African HBC environment

HBC programmes in South Africa primarily serve poorer communities within informal settlements where the need is greatest. Careworkers are often members of the community,

[2] Some examples: Microsoft Map Point (commercial), Concorde (open source, www.tsp.gatech.edu), OsmSharp (open source, http://sourceforge.net/projects/osmsharp/). Also refer to the Vehicle Routing Software Survey [11].

specifically trained to address beneficiary needs. For the most part careworkers do not have private transport and are dependent on public transport (for example minibus taxi's), walking or cycling to perform home visits. Typically these careworkers do not have easy access to internet services, either via smartphones or computers, and they have limited or no computer skills.

Within a HBC programme the needs of different beneficiaries require different careworker skill sets. For example an OVC household with a terminally ill parent may require basic medical care in addition to emotional and social support. There is thus preferential assignment of careworkers to beneficiaries. In addition, it is preferable that assignments not change often as it is important for the beneficiary to 'see a familiar face'. [1, 3, 6, 8, 9, 10]

Time windows also come into play as careworkers have certain working hours and beneficiaries may have preferred visiting times. In the case of OVC programmes, visits can only be made in the afternoon when children are home from school [8]. HBC services that provide daily meals to beneficiaries would be constrained by meal times and how long the meals can stay warm during transit [3].

The travel times between two specific beneficiary locations may vary considerably depending on the careworker and mode of transport. Careworkers could make use of minibus taxi's if these are available on the route between the two beneficiaries. Minibus taxi's do not follow specific schedules or routes and stop for passengers literally anywhere along the road. This makes travel times by taxi considerably variable, albeit faster and less tiring. Walking and cycling are also options for careworkers. When walking or cycling through informal settlements one does not always have to stick to recognised routes as there are a myriad of back-alleys and cut-throughs, the use of which would depend on the careworker.

Du Plessis [8] studied the case of a particular OVC care programme active in the Nellmapius semi-urban settlement east of Pretoria. Nellmapius' estimated population size is 65 000 with 132 households served by the OVC programme. The careworker to household ratio was 1:26, far exceeding the ideal ratio of 1:10. It was found that careworkers were not able to adequately plan their daily home visits, the result being arriving at an empty house outside of the preferred time windows or neglecting home visits on the outskirts of the settlement because not all the home visits could be concluded in a day.

3 Solution approaches for routing and scheduling in various HBC environments

HBC is a popular alternative to hospitalisation and/or institutionalisation in many developed countries as well. Home based health care (i.e. nursing patients in their homes) is a growing business sector as it is more cost effective and results in a better quality of life for patients[1, 6]. In countries such as Sweden the aging population is making HBC for the elderly an attractive option to curb governments' welfare burden [10]. As a result, a number of OR studies have sought to improve the operational efficiencies of HBC.

The LAPS CARE system [10] has been successful in significantly reducing both planning time and travelling time in ten HBC organisations in Sweden. The system is an integrated software package containing a map repository for calculating internodal travel times, an SQL database containing client and staff data, a shortest path module, an optimisation module and various GUI interfaces. Although significant time is required to initially register all the clients and staff in the database, the system can provide a good solution for the day's home visits in minutes. The HBC problem is modelled using a set partitioning model and solved with a repeat matching algorithm. Although both the nurse rostering and trip routing aspects are incorporated into the model, more emphasis is placed on matching staff members to the right clients. An unanticipated benefit of better planning in HBC organisations was the improvement of working conditions for the nursing staff and a resulting morale boost.

Bertels and Fahle [6] describe the PARPAP software, aimed at private medical service providers that run home health care services. They propose a hybrid approach that solves the nurse rostering and trip routing problem interdependently, using a combination of constraint programming, linear programming and metaheuristics. The software is a generic tool that provides the flexibility to customise many hard and soft constraints relating to time windows and patient-nurse matching. It is a highly sophisticated model that utilises various combinations of solution approaches to develop alternative solutions from which the end-user can choose.

The development of both the PARPAP and LAPS CARE software accommodated the fact that end-users in the HBC sector do not fully trust a single, machine generated solution and prefer to be given a choice between solutions and be allowed to make manual changes to the solution based on their judgement.

Other solution approaches that have proved effective in addressing the scheduling and routing of HBC are: particle swarm optimisation [1]; a combination of mixed integer programming and a heuristics approach [7]; a combination of a variable neighbourhood search and tabu-search [8]; and even simple scheduling heuristics [5].

Any of the successful solution approaches mentioned could sufficiently solve modelled representations of the HBC problems that arise in South Africa. (In fact, Du Plessis [8] focuses specifically on the South African context.) However, there is a disconnect between the research and the implementation. From a data point-of-view it would be difficult to get accurate internodal distances/travel times for homes in settlements due to the variability of travel modes and the fact that mapping software does not account for potential short cuts that deviate from the road network (see Figure 1). Figure 1 shows how the (purple) route determined by Google Earth for locations in a settlement in South Africa is much longer than the short-cuts that could potentially be taken.

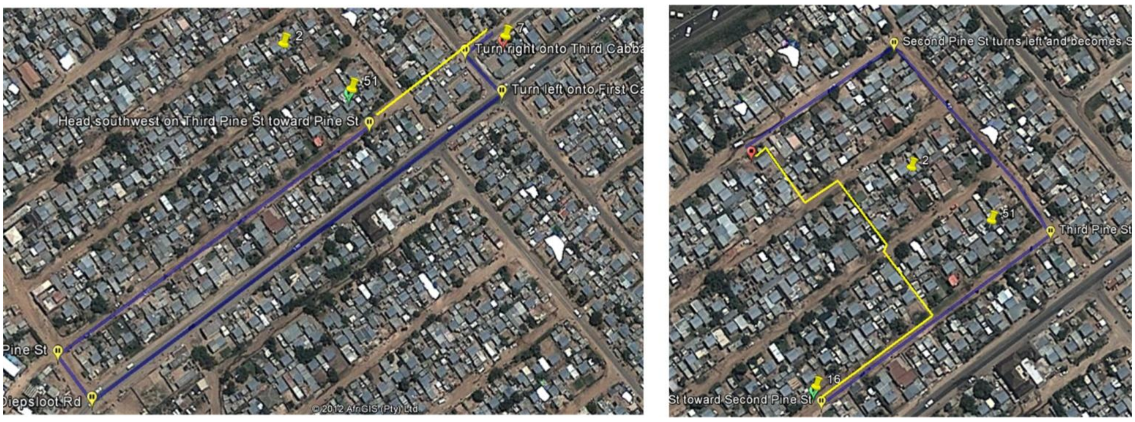

2
7
Turn right onto Third Cabba
51
Turn left onto First Ca
Head southwest on Third Pine St toward Pine St
Pine St
Diepsloot Rd
©2012 AfriGIS (Pty) Ltd
Second Pine St turns left and becomes S
2
51
Third Pine St
16
St toward Second Pine St

Figure 1: *Mapping software cannot account for potential short-cuts in settlements in South Africa*

More important than the data issue would be the technology gap. Even if the software could be installed at headquarters and a staff member could be spared to learn and use the software on a daily basis, real-time connectivity between headquarters and careworkers in the settlements is not yet a reality. This impedes the exchange of data about the daily changes to the beneficiary list and volunteer absenteeism as well as the transmittal of the home visit solution to the careworkers. Currently, for a solution approach to be implementable it needs to be executable by the careworkers within the settlement. The spacefilling curve heuristic is a more realistic solution approach for the South African HBC context as it can even be executed without a computer, using index cards and a Rolodex [3].

4 The spacefilling curve heuristic for TSP applications

Bartholdi and Platzman were the first to apply fractal geometry to combinatorial optimization in the 1980s by developing a heuristic based on the concept of spacefilling curves [2, 4, 14]. In 1982 they introduced the spacefilling curve heuristic for the planar travelling salesman (TSP) problem showing that it is easy to implement, fast to execute and produces solutions of acceptable quality [2], especially when compared to other simple TSP heuristics such as nearest neighbour, minimum spanning tree and strip [14]. Later they extended their work to other combinatorial problems in Euclidean space [4]. Bartholdi et al. [3] describes a successful implementation of a spacefilling curve heuristic based on the Sierpinski curve to plan delivery routes for the Meals-on-Wheels community service programme in Atlanta, Georgia. Recently, the same methodology was followed to develop a home delivery solution for Meals-on-Wheels in Ohio [17].

4.1 Characteristics of the generic spacefilling curve heuristic

The description of spacefilling curves dates back to the turn of the twentieth century with mathematicians like Peano, Hilbert and Sierpinski [4]. Bartholdi and Platzman [4:292] explain a spacefilling curve as *"...the limit of a sequence of recursive constructions whereby the square is subdivided into smaller squares, into which are copied scaled versions of the preceding construction."* Figure 2 shows the first six iterations of the Hilbert curve.

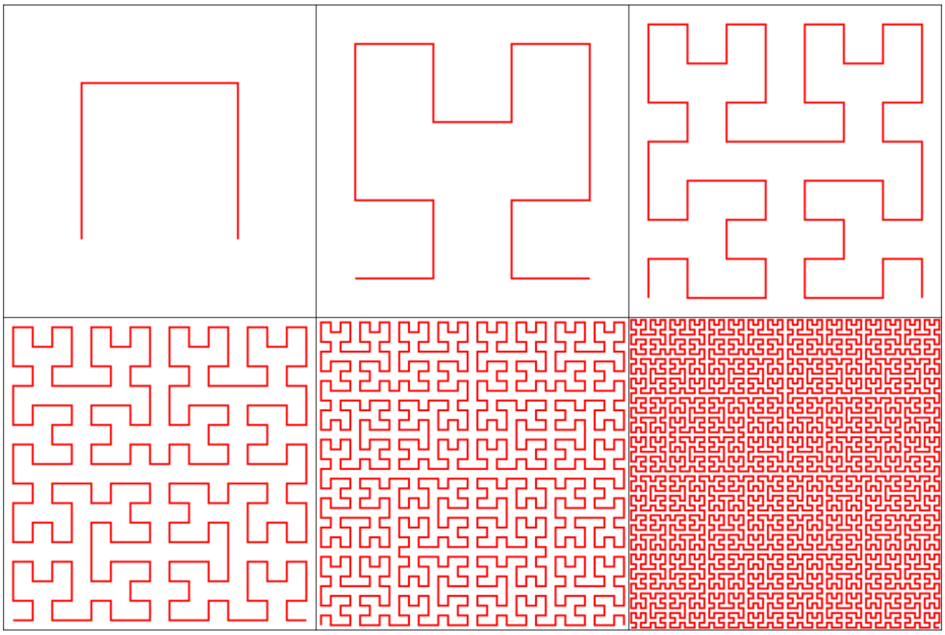


Figure 2: *Six iterations of the Hilbert spacefilling curve [18].*

If one imagines the spacefilling curve as a piece of string, a simplified analogy of the heuristic is as follows:

1. Using the string, create a spacefilling curve in a square that encapsulates all the nodes of the TSP, starting from the southwest quadrant;
2. Glue each node to the piece of the string that corresponds with the node's position in the square;
3. Pull the string taut;
4. The routing sequence is given by the order in which the nodes appear on the straight piece of string.

A complete description of how to map nodes in the unit square onto the unit interval via spacefilling curve (steps 1-4 above) is documented in Platzman and Bartholdi [14].

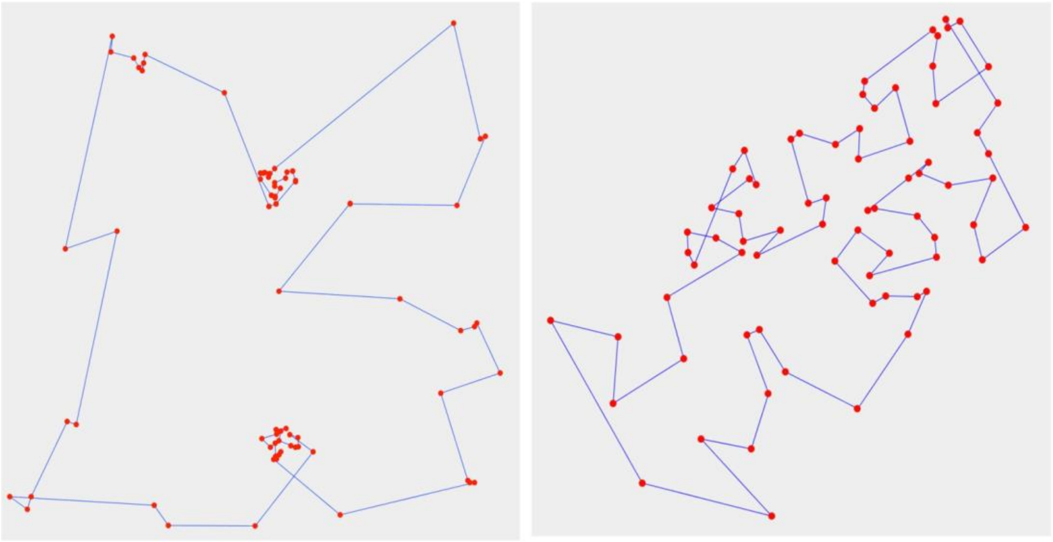
The performance of the generic spacefilling curve heuristic is analysed in detail in Platzman and Bartholdi [14] and summarised here. Regarding computational complexity it is extremely fast with $O(n \log n)$ steps in the worst-case and $O(n)$ steps in the expected case. The worst case heuristic tour length is $2\sqrt{n}$ and the ratio of the (heuristic tour length/optimal tour length)$\leq O(\log n)$. For independent, identically distributed nodes the tour length is roughly 25% longer than the optimal for large n. The heuristic is frugal in terms of data requirements, only requiring the $O(n)$ node locations and not the $O(n^2)$ intermodal distances. Finally, when adding nodes it is not required to resolve the entire instance, only the position of the new node on the unit interval is calculated and the node inserted into the sequence accordingly. Similarly, removing a node does not require resolving the problem.

The heuristic also preserves nearness of nodes as nodes that lie close together on the unit square typically lie close together when mapped to the unit interval via a spacefilling curve. However, for this characteristic to translate into a real-world TSP instance requires that customer locations that are near each other on the unit square also need to be near each other in terms of internodal travel distance, implying a pervasive transport network that creates a well-connected graph.

Following on from the nearness of nodes, problem instances with uniformly distributed nodes result in consistent internodal (Euclidean) distances. This means that if the route is divided into multiple sub-routes, each sub-route containing an equal number of nodes, the sub-route tour lengths would be roughly balanced.

4.2 Examples of tours generated by the Sierpinski spacefilling curve heuristic

Figure 3 shows tours produced by the Sierpinski spacefilling curve heuristic for two distinct problem instances. The first tour is a delivery route calculated during the project conducted for Meals-on-Wheels in Ohio, USA [17]. The distribution of beneficiaries is non-uniform, mostly concentrated in the small towns spread out across the rural counties of Logan and Champaign. The second tour routes home visits in a section of the Diepsloot semi-urban settlement, northwest of Johannesburg, South Africa. The Diepsloot dataset was constructed for illustrative purposes. Beneficiaries are more uniformly distributed among the dense residential blocks of this settlement.



Meals-on-Wheels deliveries, Ohio, USA

Home visits, Diepsloot, South Africa

Figure 3: *Two tours produced by the Sierpinski spacefilling curve heuristic [12].*

5 Appropriateness of the spacefilling curve for South African HBC

5.1 Computerless implementation of the spacefilling curve heuristic

Implementing the spacefilling curve using index cards and two Rolodexes instead of a computer has worked successfully before [3]. On the index cards the names, addresses and service requirements of the beneficiaries are written. One Rolodex orders these cards in alphabetical order and serves as a client database of sorts. To execute the heuristic, a large map of the area and a 100x100 perspex grid is required. Placing the perspex grid over the map, one can pinpoint in which block a beneficiary location lies. Using a table of precalculated values one looks up the spacefilling curve index for that location. In this manner one creates a duplicate set of index cards, this time adding the spacefilling curve index in the top right corner. The duplicate set of index cards is then ordered according to the spacefilling curve index in the second Rolodex, which now represents the master route.

Apart from the effort required to initially create the two Rolodexes, adding or removing beneficiaries is easy. To add a beneficiary, create two identical index cards, placing one in alphabetical order in Rolodex . On the other note down the spacefilling curve index before inserting it in Rolodex 2 according to its index. That's it! No need to reshuffle or resort Rolodex 2. Removing a beneficiary is as easy as removing both index cards. This is a very attractive characteristic of the system as beneficiary lists are volatile.

Partitioning the master route into sub-routes is easily done by segmenting Rolodex 2 using paperclips. Partitions can be made to include a similar number of beneficiaries in each sub-route or according to any other partitioning scheme, as long as index cards in Rolodex 2 are kept in order. This flexibility makes it easy to deal with absenteeism among careworkers.

Bartholdi et al. [3] observed that the most significant violation of the system is when the manager does not look up the spacefilling curve index for a new client, but instead "eyeballs" the position on the map and inserts it into Rolodex 2 accordingly. Presumably this practice is not tragic if done once or twice – say when the manager is in a hurry - as the index cards can always just be taken out and reinserted correctly at a later stage. However, if gone unchecked this practice would degenerate the routing solution. Despite this and other minor violations by drivers, Meals-on-Wheels in Atlanta reported that their average driving time to deliver 200 meals daily was reduced by 13%.

5.2 Addressing preferential assignment

The spacefilling curve heuristic does not account for preferential assignment as-is. From the various studies referenced earlier it is clear that preferential assignment is critical to service delivery when nursing activities are involved. In for-profit HBC [6], preferential assignment is critical for client retention as well. But the HBC environment in South Africa differs slightly. The skill set required for many HBC beneficiaries is not highly specialised (i.e. few home visits require nursing or occupational health services) as the focus is mainly on OVC households. In addition, given the dire need of most beneficiaries, less emphasis would be placed on personal preferences. Nonetheless, language, gender, and levels of training must always be accounted for when assigning careworkers to beneficiaries.

Finding a way to incorporate preferential assignment into the spacefilling curve approach will require working with various HBC providers to understand their specific constraints. Presumably the quickest way would be to develop rules of thumb for different organisational setups, but easy-to-implement, adequately accurate heuristics may soon emerge.

6 Conclusion

HBC in South Africa is an important service delivery model that reduces the burden on the country's health and welfare systems. Although caring for OVC households is the primary focus of most HBC programmes, HBC may also include services to the ill and elderly. A number of successful solution approaches have been developed over the past two decades to address the scheduling and routing of home visits in various HBC environments. Unfortunately, the data and technology requirements of these solution approaches make implementation within South Africa's HBC environment currently unrealistic. The spacefilling curve heuristic is an agile and easy-to-implement routing algorithm that requires far less data and can be implemented without a computer. This solution approach will go a long way in creating more efficient daily home visit plans. Working with organisations as they implement the spacefilling curve solution approach, mechanisms can be developed to deal with preferential assignment in each setting.

The author has approached four NGO's during the preparation of this paper to assist them in implementing the spacefilling curve solution approach to improve the HBC and other routing problems unique to their organisations. Although these NGO's were eager to share their operational headaches and it could be determined that improved routing would indeed make their services more efficient, successful implementation has not yet been possible. One reason for this is that NGO's do not believe, up front, that the potential time and cost savings would offset the time investment and risk of failure from their side, especially when increasing routing efficiency is near the bottom of the priority list. The other reason is that internodal distance and travel time datasets are not readily available for informal settlements. Therefore, algorithm solutions cannot be easily translated into actual distance or travel time for comparative purposes. The way forward can adopt a two-pronged approach: while efforts are made to produce convincing results for South African HBC cases using the Nellmapius dataset [8], manual data collection and/or platforms like OpenStreetMap, practitioners can work with NGO's on unrelated operational problems to gain trust and "learn-while-doing" so that eventually practitioners can be entrusted with implementing the routing solution.

Bibliography

[1] Akjiratikarl, C., Yenradee, P., and Drake, P.R. (2007). PSO-based algorithm for home care worker scheduling in the UK. *Computers and Industrial Engineering*. 53:559-583.

[2] Bartholdi, J.J. III, and Platzman, L.K. (1982). An $O(N \log N)$ planar travelling salesman heuristic based on spacefilling curves. *Operations Research Letters*. 1(4):121-125.

[3] Bartholdi, J.J. III, and Platzman, L.K. (1988). Heuristics based on spacefilling curves for combinatorial problems in Euclidean space. *Management Science*. 34(3):291-305.

[4] Bartholdi, J.J. III, Platzman, L.K., Collins, R.L., and Warden, W.H. III. (1983). A minimal technology routing system for Meals on Wheels. *INTERFACES*. 13(3):1-8.

[5] Begur, S.V., Miller, D.M., and Weaver, J.R. (1997). An integrated spatial dss for scheduling and routing home-health-care nurses. *INTERFACES*. 27(4):35-48.

[6] Bertels, S., and Fahle, T. (2006). A hybrid setup for a hybrid scenario: combining heuristics for the home health care problem. *Computers and Operations Research*. 33:2866-2890.

[7] Cheng, E., and Rich, J.L. (1998) A home health care routing and scheduling problem. Technical report CAAM TR-98-04, Rice University.

[8] Du Plessis, W. (2010). A metaheuristic approach to the assignment, scheduling and routing of care workers in the home and community-based scenario in South Africa. Final year dissertation, University of Pretoria, South Africa. Available: http://repository.up.ac.za/handle/2263/16533.

[9] Du Plessis, W., Bean, W., Schoeman, C., and Botha, J. (2011). *OR/MS Today*, April. 38(2). Available: http://www.informs.org/ORMS-Today/Public-Articles/April-Volume-38-Number-2/Home-and-community-based-care-in-South-Africa.

[10] Eveborn, P., Flisberg, P., and Rönnqvist, M. (2006). LAPS CARE – an operational system for staff planning of home care. *European Journal of Operational Research*. 171:962 – 976.

[11] OR/MS Today.(2012). Vehicle Routing Software Survey. February, 2012. Available: http://www.orms-today.org/surveys/Vehicle˙Routing/vrss.html. [Cited 28 June, 2012].

[12] GoogleMaps. Available: www.maps.google.co.za. [Cited 28 June, 2012].

[13] Peu, M. D., Tshabalala, A. M., and Hlahane, M. S. (2008). Home/community-based care. Van Schaik.

[14] Platzman, L.K., and Bartholdi, J.J. III. (1989). Spacefilling Curves and the Planar travelling Salesman Problem. *Journal of the Association of Computing Machinery*. 36(4): 719-737.

[15] Rahman, S., and Smith, D.K. (1990). Is 'appropriate OR' necessarily 'simple OR' for developing countries? *OPSEARCH*. 27(4):264-268.

[16] Strebel, A. (2004). The development, implementation and evaluation of interventions for the care of orphans and vulnerable children in Botswana, South Africa and Zimbabwe. HSRC Publishers, Cape Town. Available at: www.hsrcpublishers.ac.za.

[17] Tri-County Community Action Commission. (2012). Georgia Tech lends a hand to meal routes. *The Inside Look, CLS Employee & Volunteer Newsletter*. 2(1):1.

[18] Wikipedia. Space-filling curve. Available at: http://en.wikipedia.org/wiki/Space-filling˙curve. [Accessed: 29 June 2012.]

[19] Woolsey, R.E.D., and Swanson, H.S. (1975). *Operations Research for Immediate Application*. Harper and Row. New York.

Proceedings of the 2012 ORSSA Annual Conference
pp. 67–77
www.orssa.org.za/wiki/uploads/Conf/2012ORSSAConferenceProceedings.pdf

ORSSA Proceedings
ISBN
978-07972-1406-4
© 2012

Improving the work rate of community health workers through optimisation

FJ Snyders[1] TE Lane-Visser[2]

Abstract

The aim of this paper is to show that an optimised route can reduce the travel time and cost required for community health workers (CHWs) to perform their work and to get a sense for the magnitude of potential saving when applied to a realistic case study. Through optimised route planning, CHWs can achieve higher work rates, resulting in more effective resource utilisation. One of the tasks of CHWs is capturing health related data. Technology (such as EpiSurveyor) exists that improve the efficiency of capturing data through using mobile phones. The problem lies in that CHWs have to visit many households to capture data across large distances, resulting in high travel costs and much time spent travelling. In this paper the CHWs' situation is modelled as a travelling salesman problem (TSP). The paper shows how optimised route planning can be used to reduce the time and cost of travel by 26%, resulting in budgetary savings and a reduction in the required man-hours to administer a given set of surveys.

Key words: Travelling Salesman, Health services, Heuristics, Routing problems, Tabu search.

1 Introduction

With the increased utilisation of mobile phones in remote areas, an opportunity arises for improving health services in remote areas. In South Africa alone 46,9% of the population are active data users under Vodacom (58% market share) and MTN (35.5% market share) [1] [2]. Areas that could not access health services because of their remote location can now access certain health services through mobile phones. The use of mobile phones for improving health and wellness is known as mHealth [3]. These health services include the following [3]

1. Education and Awareness
2. Remote Data Collection
3. Remote Monitoring
4. Communication and Training of Healthcare Workers
5. Disease and Epidemic Outbreak Tracking

[1] Corresponding author: Department of Industrial Engineering, Stellenbosch University, Private Bag X1, Matieland, 7602, South Africa,, email: franslight@gmail.com
[2] Department of Industrial Engineering, Stellenbosch University, Private Bag X1, Matieland, 7602, South Africa,, email: tanyav@sun.ac.za

6. Diagnostics and Treatment Support

Remote data collection is one of the applications of mHealth that has been widely implemented. Health surveys fall under the category of remote data collection. Health surveys play an important role in monitoring the status of a region's health. Data from health surveys can be used to monitor the health of a population or to identify bottlenecks in a health system. Areas can be targeted to receive additional resources and attention, in order to improve the general state of health of a country. If data is continually collected over time, historic data can be used to monitor and measure the effect of health initiatives on the population's health.

Collecting data via a mobile phone has many advantages over a paper based system, most notably is the shorter data processing time [4]. One of the most prevalent tools in the health field for such surveys is EpiSurveyor [5] [6]. With EpiSurveyor data entry forms (such as health surveys) can be created, filled in, and submitted using a mobile phone. The captured data is then automatically analysed and can be accessed via other mobile phones or computers. These results are used for reporting in order to monitor and improve the state of health care in a region.

Many parties are involved in the collection of health information in a country. These parties can either be private or state funded and local or international. For the purpose of this paper, the term community health workers (CHWs) will be defined as any party or person that is funded by the national health system with the task of collecting health information from the population.

By reducing the time taken to complete the task of collecting health information, other necessary health related tasks can be addressed by CHWs. The responsibilities of CHWs include managing water supply, first aid and treatment of simple and common ailments, provision of health education, treatment of acute respiratory infections, and communicable disease control, amongst others [7].

Even though mobile phone based surveying decreases the time CHWs need to process surveys after completion, the requirement still exists to visit households in person in order to collect certain health related data. These visits to houses take up much time in terms of travelling between households. Routes are usually planned manually and it is expected that if route planning software could be used, time and money could be saved.

Reduced travel costs in completing health surveys can result in smaller budget requirements for monitoring a country's health. This also frees up resources (CHWs) to address other needs identified. Because of skill shortages in the African context, CHWs have an increased workload, resulting in less time available for the collection of health data. The ultimate result can be improved health service delivery in the country.

2 Problem description

Brunskill and Lesh [8] highlighted the need for the use of a routing algorithm by CHWs that need to visit multiple households in rural locations. This paper is an attempt to apply such an algorithm to the CHW context, in order to increase the work rate of CHWs. To evaluate the impact of route optimisation in a CHW's context, a case study will be used. This is done to improve the realism of estimating the magnitude of possible time and cost savings.

In order to provide a specific context for the problem, the 2010 Kenya malaria indicator survey [9] is used. The malaria indicator survey used a representative sample of 7200 households taken across Kenya. Each district in Kenya was assigned a number of clusters, according to the population of the district. This resulted in a total of 240 clusters for Kenya. Each cluster consisted of 30 households that were randomly chosen from a database of houses in the cluster with the use of personal digital assistants (PDAs). The PDAs had the ability to record the GPS coordinates for the households where data was collected. The survey consisted of a list of demographic and health related questions, which took 15-20 minutes each to complete [9].

One of the districts in Kenya where the survey was conducted is called Embu. The location of Embu in Kenya can be seen in Figure 1. In this survey, Embu was allocated two clusters (60 households) [9]. One of these clusters (the northern cluster) was used as a basis to generate 30 random households. The northern part of the Embu district can be seen in Figure 1. The 30 households that were randomly selected in the region can be seen in Figure 2.

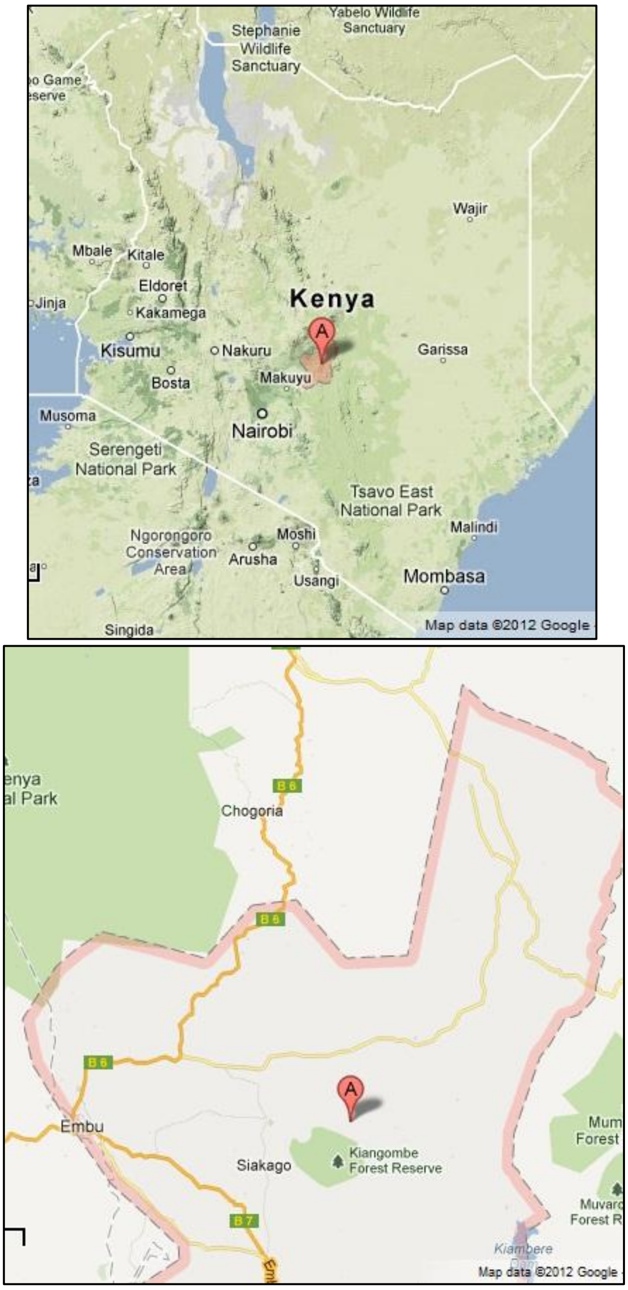


Figure 1: *Map showing Embu's location in Kenya (A), and the northern part of the Embu district*

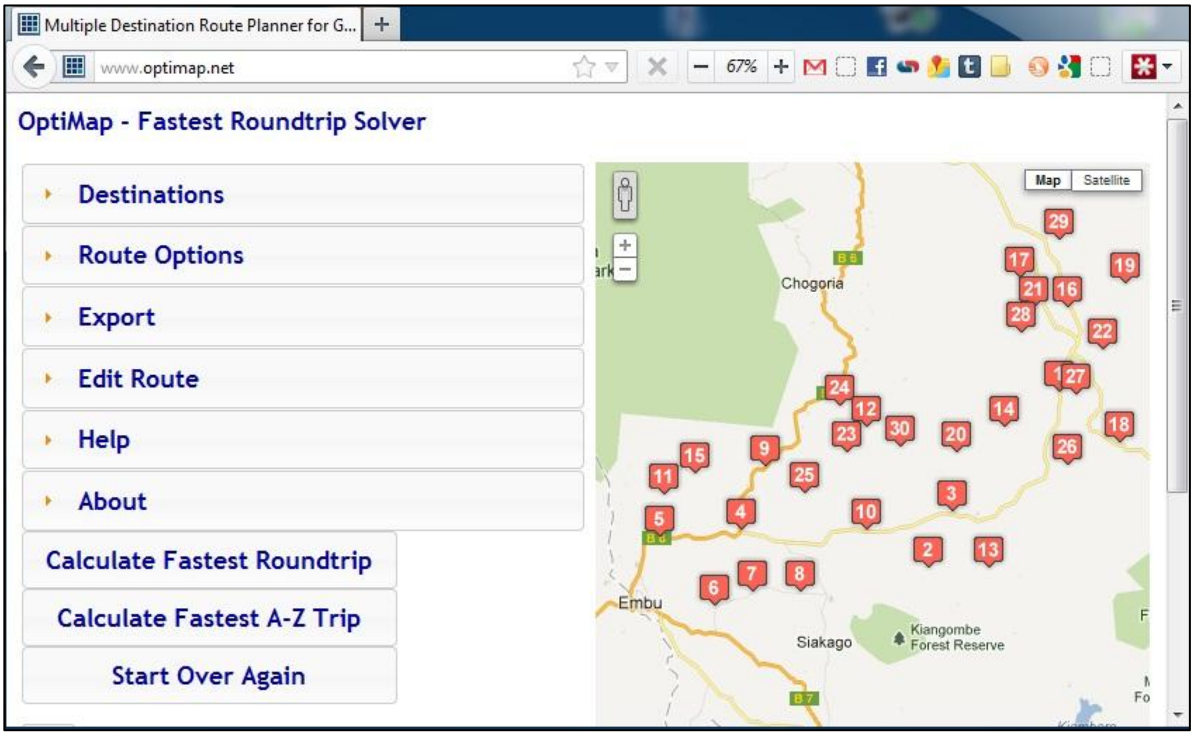


Figure 2: *Screenshot of OptiMap[15] showing the randomly chosen households*

The problem can be stated as the determination of the route that a CHW has to follow in order to collect health data from multiple households efficiently. This problem can be reduced to a travelling salesman problem (TSP), where the objective is to find the shortest route that traverses all the households that need to be visited and then returns to the starting household at the end of the tour.

The aim of this paper is to show that an optimised route can reduce the travel time and cost required for CHWs to perform their work and to get a sense for the magnitude of potential savings when applied to a realistic case study. The optimised route and resulting travel time and cost will be calculated through an optimisation model (to be discussed in Section 3).

2.1 Factors to be considered

Many factors affect the routing of the data collection tasks of CHWs. The factors considered in this study are listed in **Table 1**.

Table 1: *Factors affecting route choices as listed by Brunskill & Lesh [8]*

	Factor	Description
1	Distance	The distance between the households.
2	Travel speed	The average travel speed travelled between households, depending on the mode of travel (e.g. walking/taxi). From the average travel speed and distance data, the average travel time can be calculated.
3	Travel cost	The cost incurred to travel between two households. This also depends on the mode of transport used.
4	Visit time	Time taken to complete a house visit. This depends on the time it takes

		to collect data.
5	Revisit	The need to revisit a household can occur due to the absence of an occupant, incorrect data entered, or when more frequent data collection is required.
6	Work hours	The time window in which a CHW can collect data. This can depend on daylight hours, the weather, and the availability of the community, the current work load of the CHWs or other resource logistics.
7	Priorities	Some tasks may require the CHWs' attention above that of data collection, depending on the workload of the CHW.
8	Occupants	The number of people living at a house. This can increase visit time, due to more data that needs to be entered.
9	Logistics [8]	Availability of resources, including mobile phones, signal, and electricity to charge phones.

3 Modelling

As stated before, the objective of the model is to find the shortest route that traverses all the required households and then returns to the start household at the end of the tour. The result of the model is the sequence in which households should be visited to form an optimal route. CHWs can use the generated sequence to know in which order to visit the households, and thus minimise their travel distance and time. This section describes how the factors in Table 1 are implemented in the model.

3.1 Model Parameters

The distances between houses were calculated from the physical paths between these households. The data was obtained by using Google Mapsff, resulting in a complete distance matrix for the households in question.

The average travel speed is taken as the speed of driving in a residential area (60 km/hr). It is assumed that CHWs have access to a car. In reality, the travel speed may be greater if conditions allow this. Contradictory, some roads travelled on can be dirt roads, have potholes, or an incline, which would result in a reduction in average travel speed. The elevation difference between the random points in Embu can be up to approximately 1400m. It is assumed that the model will be used in rural areas, thus access to straight tarred roads will be limited. For the purpose of this paper, the model is limited to a fixed average speed for all roads travelled, although it can be expected that the travel times will be underestimated to some extent.

The average travel time can be calculated from the distance travelled and the speed, as defined.

It is assumed that the cost of transport is directly proportional to the distance travelled. The actual cost differs for each country, and the cost is assumed to be a constant c in each country. In reality, some CHWs may use various public transport options to visit all the households that have to be surveyed. An adapted average cost per distance travelled can be used in these situations.

The cost used in this paper is calculated assuming a light vehicle is used for transport. The average fuel consumption is assumed to be around 7.0 l/100km, with associated vehicle depreciation of R 0.10 /km. The fuel price in Kenya was R10.88 on 30 June 2012. This gives a total cost of R 0.831/km.

The visit time is based on the time that is allocated to complete a survey (15-20 min). An additional 5 minutes is allowed to prepare for the survey and incidentals. This includes the time to locate the person that will complete the survey. An extra 5 minutes is also added after the survey, to give time for CHWs to locate the next household to be travelled to. The actual time may vary according to the ease of finding a person at the household to complete the survey. The additional 5 minutes before and after the survey is only an estimate, because no exact time stamps could be found regarding the administration of surveys in rural areas.

The need to revisit a house can arise when no suitable person is present at the household to complete the survey. In the Embu scenario, surveyors were sometimes instructed to revisit a household up to three times [9]. For the random selection of houses, contact details were not known. If contact details were known, surveyors could call ahead of time to ensure that a suitable person will be available at the house.

The problem in modelling a revisit is that there is no way to predict which house will generate a revisit before it has been visited in reality. If a model was to predict three revisits to a household, yet no revisit is generated in reality, then the resulting sequence of the route may be less optimal because of the consideration of the revisits. Alternatively, if a revisit is not predicted, yet occurs, the resulting sequence will always be less than optimal. So, although there are algorithms that can handle the inclusion of revisits in the original route, these will not be useful in this case. The basic routing algorithm will have to be rerun as soon as knowledge of a revisit is obtained.

The work hours of CHWs are the time window in which CHWs can perform data collection. It is assumed that CHWs will administer the health survey by following the sequence of routed households until their work hours (around 9 hours per day) have ended. If CHWs are unable to recalculate their route the following day, it is also assumed that CHWs will pick up surveying from where they left off the previous day and continue with the rest of the sequence. This introduces further travelling expenses as CHWs travel to and from the location where they sleep over.

The order of priorities of CHWs changes over time, depending on tasks that have to be completed. It is assumed that the CHWs' first priority for the day is the completion of the surveys. It is assumed that no other tasks will interfere with the CHWs schedule of taking surveys.

The number of occupants in a household has a minor impact on the visit time, seeing that more data fields may not have to be filled in on the survey, but rather different data. An example is the recording of the number of occupants, and when last a visit was made to the local clinic. It is assumed that the average time posted to collect data incorporates any variance in the time spent taking the survey.

It is assumed that all logistical challenges of CHWs have been addressed prior to the day of administering the survey. If mobile phones have no signal or lose power, surveys can be done on paper as a backup, whereafter it can be electronically captured at a later stage or by someone else.

3.2 Nearest neighbour search

Seeing that no data is available regarding the specific route surveyors took in Embu, a route was approximated by using the nearest neighbour search algorithm. Tours constructed by the nearest neighbour algorithm frequently falls within 25% of the Held-Karp lower bound. The Held-Karp lower bound is used for problems where the optimal solution is not known [10]. This method may overestimate the efficiency with which CHWs chose routes, considering their lack of data concerning the distance between households, together with the unstructured way in which points can be chosen.

The nearest neighbour search is a method of constructing a tour by moving to the next household that is currently the most advantageous [11]. This implies that the household closest to the current household will be chosen as the next household. For this problem, the search will always start at the first household, being the community health centre (the point of departure).

In order to identify the closest household, the distances from the current household to all feasible locations are compared. Households that have been visited, including the current household, are added to a Tabu list. The procedure is repeated until all households have been visited. Finally a move from the last household to the first household is included to complete the tour.

3.3 Solving the Travelling Salesman Problem

3.3.1 Variable definition

The following input variables are defined for the model:

Let XY_i be the GPS coordinates for each house i to be visited, with $i = \{1,2,..,n\}$ and n being the number of houses to be surveyed. This is made up of the set of the longitude and latitude coordinates. Let T_i be the visit time in which a survey is completed at house i. Let s_{ij} be the average speed at which CHWs travel between households i and j. Let c_{ij} be the cost per distance travelled from household i to j.

The output variables of the model are defined as follows:

Let the tour consist of the cyclic permutation $\boldsymbol{\pi}$ with π_i the household that follows household i in the tour. The sequence that π forms is the sequence in which the households need to be visited to form an optimal route. Let the total distance travelled for the optimal tour be D_T [12].

The model objective is to determine the feasible sequence in which households must be visited in order to minimise the total distance travelled by CHWs. A shorter distance travelled results in less time spent (man-hours) on data collection, and a reduction in travel cost.

3.3.2 Travelling Salesman Problem (TSP)

According to Hahsler & Hornik [12] the goal of the TSP is to find the shortest tour that visits each city (household) in a given list exactly once and then returns to the starting city (household). The TSP can be described as follows [12]: The distances between households are stored in the distance matrix D with the diagonal elements $d_{ii} = 0$. It is further assumed that the distances travelled are symmetrical in that $d_{ij} = d_{ji}$. The cost c_{ij} is also assumed to be symmetrical. The travelling salesman problem is the optimisation problem of finding the sequence π that minimises the total length of the tour given by

$$\sum_{i=1}^{n} d_{i\pi_i} \qquad (1)$$

For this optimisation problem $(n-1)!$ tour lengths exists. This makes the problem hard to solve, with it being classified as an NP-complete problem [12].

3.3.3 Ant Colony Optimisation (ACO)

The use of a heuristic to solve the problem was chosen firstly on the basis that heuristics are known for shorter calculation times [13]. Shorter calculation times for heuristics come at the price of obtaining near optimal results, instead of the exact solution [13]. For this problem, it is assumed that a near optimal solution will be acceptable. Secondly, a heuristic approach enables the optimisation program to be scripted into a web-platform. This removes the need for additional mathematical software packages to solve the problem.

A classic algorithm named the Ant System (AS) [14] was used for heuristically solving the TSP. The AS was the first ant algorithm to be applied to the TSP [14]. As the name suggests, ant algorithms are inspired by how ants use pheromones to construct paths (trails) to food. All ant algorithms can be placed under the category of ant colony optimisation (ACO) algorithms.

Although many ACO algorithms exist [14], only the Ant System algorithm is used for this problem.

The pseudo-code for all ACO algorithms consists of repeating the following three phases (until a termination condition is met): a solution construction phase, a local search phase (optional), and a trail update phase. During the solution construction phase, probabilities are calculated for an ant to move from its current household to feasible alternative household. In this problem ants always start at the first household. The community health centre can be regarded as the first house. The probabilities for an ant to move from i to j are generated by Equation 2 [14].

$$p_{ij} = \frac{[\tau_{ij}]^{\alpha} \cdot [\eta_{ij}]^{\beta}}{\sum_{l \in N_i} [\tau_{il}]^{\alpha} \cdot [\eta_{il}]^{\beta}} \qquad (2)$$

The variable p_{ij} gives the probability of moving from i to j. The variable $\tau_{ij} : \{0,1\} \in \mathbb{R}$ is the level of the pheromone on the trail. η_{ij} indicates the vision of the ant and is given by $1/d_{ij}$. N_i is the set of points that the ant may visit from point i. This set incorporates a Tabu list, to ensure that no households that have already been visited are revisited. α and β are parameters that affect the weight of the pheromone and vision components when calculating the probability of a move.

Based on these probabilities, a move is made to the next feasible household. The probabilities to move from the new point are recalculated. The process is repeated until a tour is completed, resulting in one solution that has been constructed. Following the solution construction phase, the trails are updated by changing the pheromone level for each arc. If a given arc has a high pheromone level, there is a higher probability that routes will be constructed using the specific arc.

At the end of each iteration all the pheromone levels are reduced; thereafter the pheromone levels of the best solution to date's arcs are increased (see Equations 3 & 4). [14]

$$\tau_{ij} = (1-\rho) \cdot \tau_{ij} + \Delta\tau_{ij} \qquad (3)$$

$$\Delta\tau_{ij} = \begin{cases} Q/D^* & \mathit{if\ arc(i,j)\ is\ used\ in\ the\ best\ solution} \\ 0 & \mathit{otherwise} \end{cases} \qquad (4)$$

In Equation 3, ρ is known as the pheromone evaporation constant, indicating how fast pheromone levels decrease. The variable $\Delta\tau_{ij}$ indicates the amount of pheromone that is added. In Equation 4, Q is the pheromone addition constant, indicating the strength of new pheromones added. The variable D^* is the total distance of the best route.

3.3.4 OptiMap

OptiMap is a freely accessible webpage that can be visited from any web-enabled mobile phone. This enables CHWs to access OptiMap from any location that has data reception at the minimal cost of data usage. This makes OptiMap and effective and easily implementable solution for the context of CHWs.

The ACO algorithm has been refined since its development, and many optimisation procedures have been added to it. Many TSP solvers have been developed and are available online. One such TSP solver was developed under MIT license [16] and has been implemented on the OptiMap website [17]. A screenshot of the website is shown in Figure 2.

Multiple destinations (up to 100) can be entered either via the map or via a list of GPS coordinates. The latter was used in this problem. After the household have been entered, the fastest round trip can be calculated, or the fastest A-Z route (one way). The results show a near optimal trip's total trip duration and distance. A route map is also displayed with driving instructions, including the GPS coordinates of the households.

The TSP solver used on OptiMap uses an ACO algorithm with a k2-opting tour optimisation algorithm. K2-opting is a method of improving a route, after it has been calculated with the ACO algorithm [16]. K2-opting compares the length of the current route with one where a change is made in the order in which households are visited. The swap consists of reversing the order in which two consecutive households are visited. Only if the route's length is reduced, the swap will be made permanent. This test is performed for each household, until no improvement can be made. The result is an improved route by means of a simple method.

4 Results

The distance of the route travelled by a CHW is calculated both with the nearest neighbour search and the OptiMap ACO algorithm. Using the average travel speed and the cost information as discussed in Section 3, the time travelled and the travel cost can also be calculated. The results are compared in

Table **2**.

Table 2: *Comparison of results*

Method	Distance travelled (km)	Time travelled (hours)	Difference (hours)	Travel Cost (Rand)	Difference (Rand)	Time & Cost Difference
Nearest Neighbour	454	7.567		377.27		-
OptiMap	337.5	5.625	1.942	280.46	96.81	26%

From

Table **2** it can be seen that the route produced by the OptiMap website decreases the distance required to visit all the households. This results in a time and cost saving of 26% for the routing of one cluster. This shows that the ACO algorithm can be used to reduce the travel time and cost required for community health workers (CHWs) to perform their work.

To give an indication of possible savings, the 26% saving in time and cost for this one cluster was extrapolated to the 240 clusters of the 2010 Kenya malaria indicator survey. The authors realise that this might not be accurate and that a mere extrapolation of results is probably unrealistic, yet a (currently unknown) ball park figure can be obtained from such an analysis. It was seen that a saving of R96.81 per cluster would result in a saving of R23 234 for the 240 clusters in the survey. The time saving of 1.942 hours per cluster would result in a saving of 466 man-hours. CHWs in Kenya earns about R2440 ($300) a month, for about 180 hours per month. The result is that an additional R6 318 in salaries can be saved, or new CHWs can be appointed. Additionally, the reduction in the time CHWs are needed to administer the survey results in CHWs having more time to perform other important tasks, raiding the average level of healthcare provision in the region (See Section 1).

5 Conclusions

In this paper it was illustrated that an optimised route can reduce both the travel time and cost required for community health workers (CHWs) to perform their duties. A 26% saving in both cost and time was found when applied to the 2010 Kenya malaria indicator survey. This shows how route planning software can be used to save both time and money. Platforms like the OptiMap website is ideal for use by CHWs, seeing that it is free and can be accessed through any web-enabled mobile phone. CHWs can enter the coordinates of the households they still need to visit, and be provided with a near optimal sequence in which households should be visited. There is no additional cost associated with using the website, other than ensuring that CHWs have access to web-enabled phones and data.

Regarding future work, the assumption regarding the cost of transport and the speed of transport can be relaxed and more accurate estimates can be used. The result will be a more realistic picture of the actual cost saving that an optimised route can offer for that location. It would also be ideal to compare the OptiMap route's performance to the real routes developed by the CHWs at present and to implement a pilot test of CHWs utilising OptiMap to determine their household visitation schedules.

Bibliography

[1] Vodacom Group Ltd. (2012). Integrated report for the year ended 31 March 2012. Online. Available: www.vodacom.com/pdf/annual˙reports/ar˙2012.pdf [Cited August 23rd 2012]

[2] MTN Group Ltd. (2012). Interim results for the six month ended 30 June 2012. Online. Available: http://www.mtn.com/Investors/Financials/Documents/MTN˙InterimResultsPresentation˙2012.pdf [Cited August 23rd 2012

[3] V.W. Consulting. (2009). mHealth for development: The opportunity of mobile technology for healthcare in the developing world. United Nations Foundation & The Vodafone Foundation.

[4] World Health Organization. (2011). mHealth: New horizons for health through mobile technologies. Online. Available: http://www.who.int/goe/publications/goe˙mhealth˙web.pdf [Cited June 6th, 2012]

[5] Datadyne EpiSurveyor: Mobile Data Collection Made Simple. Online. Available: http://www.episurveyor.org [Cited June 6th, 2012]

[6] Victoria, V. & Nicogossian, A. (2011). mHealth: Saving lives with mobile technology. Online. Available: http://csimpp.gmu.edu/pdfs/student˙papers/2011/Victoria.pdf [Cited June 6th, 2012]

[7] World Health Organization. (2007). Community health workers: Evidence and Information for Policy, Department of Human Resources for Health Geneva. Online. Available: www.who.int/hrh/documents/community˙health˙workers.pdf [Cited June 6th, 2012]

[8] Brunskill, E. & Lesh, N. (2010). Routing for Rural Health: Optimizing Community HealthWorker Visit Schedules. Proceedings of AAAI Artificial Intellegence for Development, AI-D10. pp. 22-24.

[9] Division of Malaria Control (DOMC) in the Ministry of Public Health and Sanitation. (2010). Kenya: Malaria Indicator Survey (MIS). Online. Available: http://www.measuredhs.com/what-we-do/survey/survey-display-385.cfm [Cited June 28th, 2012]

[10] Johnson, D.S. & McGeoch, L.A. (1997). The Traveling Salesman Problem: A Case Study in Local Optimization. Local search in combinatorial optimization. 215-310.

[11] Laporte, G .(1992).The Traveling Salesman Problem: An overview of exact and approximate algorithms. European Journal of Operational Research 59. pp. 231-247.

[12] Hahsler, M. & Hornik, K. (2006). TSP-infrastructure for the traveling salesperson problem. Department of Statistics and Mathematics, WU Vienna University of Economics and Business.

[13] Nilsson, C. (2003). Heuristics for the traveling salesman problem. Department of Computer Science, Linkoping University.

[14] Stützle, T. & Dorigo, M. (1999). ACO algorithms for the traveling salesman problem. Evolutionary Algorithms in Engineering and Computer Science. New York, NY: Wiley. pp. 163-183.

[15] Fastest Roundtrip Solver. Online. Available: www.optimap.net . [Cited June 30th 2012]

[16] TSP Solver for Google Maps API. Online. Available: code.google.com/p/goolge-maps-tsp-solver. [Cited June 30th 2012]

[17] Behind the scenes of OptiMap. Online. Available: http://gebweb.net/blogpost/2007/07/05/behind-the-scenes-of-optimap/. [Cited June 30th 2012]

Proceedings of the 2012 ORSSA Annual Conference
pp. 78–87
www.orssa.org.za/wiki/uploads/Conf/2012ORSSAConferenceProceedings.pdf

ORSSA Proceedings
ISBN
978-07972-1406-4
© 2012

Keeping it simple in a data-sparse environment: The case of donor breastmilk demand and supply in South Africa

NM Viljoen[1] M Celik[2] W Cao[2] J Swann[2] O Ergun[2]

Abstract

Donor breastmilk could potentially save thousands of neonatal lives and save millions of Rands in treatment costs annually. A facility location-allocation model will be used to develop a strategic national network expansion plan based on an existing breastmilk banking service model. The disaggregate demand and supply data required by this location-allocation model do not exist as-is in South Africa. This is often the case when developing OR models for developing countries. This paper thus discusses a simple methodology whereby the input data for the location-allocation model are prepared and not the location-allocation model itself. The methodology combines demographic data, health statistics and insights from literature and subject experts to determine that in 2011 almost 90 000 premature infants without access to Mother's-own-Milk would have required more than 1.7 million bottles of pasteurised donor breastmilk to protect them from fatal infections during the first 14 days of life. Simultaneously, 160 000 bottles of unpasteurised donor breastmilk could be sourced from potential donors. The disaggregate estimates show that supply and demand are geographically disparate and that at most 43% of demand could be covered with the given demand. This has implications for the model development, specifically in accounting for equitable distribution.

Key words: Data-sparse, facility location-allocation, humanitarian operations research, breastmilk donation

1 Introduction

Globally, 4 million neonates die annually in the first 28 days of their life [11]. Neonatal infections are among the leading causes of neonatal deaths in developing countries. The most effective intervention to reduce the risk of neonatal infections is to ensure that the infant is breastfed [10]. Furthermore, the World Health Organization (WHO) has found that in Sub-

[1] Corresponding author: CSIR Built Environment, South Africa, PO Box 395, Pretoria 0001, email: nviljoen@csir.co.za
[2] Center for Health and Humanitarian Logistics, Georgia Institute of Technology, United States of America

Saharan Africa alone, increasing breastfeeding can prevent 1.5 million infant deaths annually [7]. In South Africa, maternal death during birth, maternal illnesses such as HIV and TB-meningitis and lack of rooming-in facilities in public hospitals reduce infant access to Mother's-own-Milk (MoM), especially among premature infants. Premature infants are much more vulnerable to fatal infections than full-term infants. The South African Breastmilk Reserve (SABR) [20] has found that providing premature infants without access to MoM with Pasteurised Donor Breastmilk (PDB) for the first 14 days of life is highly effective in preventing and treating Necrotising Enterocolitis and other fatal infections. Not only does this intervention save lives, but it also saves Neonatal Intensive Care Units (NICUs) millions of Rands in treatment costs annually.

Given the high prevalence of HIV and TB in South Africa and the enduring service gaps and capacity constraints in public health care, it is clear that there are potentially many premature infants that do not have access to MoM. A number of not-for-profit organisations have been established in South Africa to provide donor breastmilk to infants. One specific organisation, the South African Breastmilk Reserve, has developed a successful service model that collects, pasteurises and re-distributes donor breastmilk equitably. The service model's long-term sustainability through Public Private Partnerships and volunteer buy-in has been illustrated in Gauteng. Capital investment is required to expand the SABR network nationally. Tabling such a proposal to government bodies or international funders requires a thorough, substantiated 'business case'. Transportation is by far the biggest cost contributor within the SABR and, in addition, it is also the most frequent cause of service failure. A network expansion plan thus has to take into account not only facility location and assignment, but also the impact of the network design on transportation cost and reliability given alternative transportation solutions (e.g. courier, volunteer transport). But this paper does not discuss the network expansion model or its results[3], it discusses how the team collated data from various sources to determine the demand for Pasteurised Donor Breastmilk (PDB) and supply of Unpasteurised Donor Breastmilk (UDB) per local municipality per year in a data-sparse environment.

In their review of location-allocation models in health service development and planning in developing nations, Rahman and Smith [19] mention that data intensive and data sensitive models are not applicable to developing nations. Harper and Shahini [8] developed a decision support system for the care of HIV and Aids patients in India that had to be mindful of the lack of data typical of developing countries. For OR to have successful application in developing countries, models and solution approaches should be very mindful of the culture and the people who will interact with the technology [18]. Similarly, for OR applications in data-sparse environments the availability, accuracy, disaggregation and currency of data sources should inform and guide model development.

2 Data requirements for the national supply and demand of donor breastmilk

The objective of this study was to quantify the capital investment required to establish a national breastmilk banking service, based on the SABR service model, through the development of a strategic facility location-allocation model. The model had to inform how many of each type of facility was required, where these facilities had to be located and how facilities would be allocated to serve the network needs. Such a model would provide ballpark budgets based on detailed modelling of facility establishment and transportation costs over a five year planning horizon. On a strategic level, it was decided that the question of what it means to service national demand equitably should not be influenced by current capacity and service constraints within the health care system. This meant that national demand had to be determined by demographics and not by the number of NICU beds available. Similarly, national

[3] These are discussed in a forthcoming paper.

supply was also determined by demographics and literature regarding donor behaviour as opposed to being extrapolated from the limited statistics available in Gauteng.

The lower the level of data disaggregation, the greater the data burden but the higher the granularity at which the model could be solved. However, if data are not available at a disaggregated level, 'expert assumptions' or even reasonable guesses have to be used – endangering the validity of the results. The fact that the lowest level of disaggregation at which population statistics are reported by Statistics South Africa (StatsSA) is the local municipality level, determined that demand for bottles of PDB and supply of bottles of UDB would also be estimated at a local municipality level.

To quantify demand, the question was: "How many bottles of PDB are required per local municipality per year?" This requires knowing how many premature infants without access to MoM would be born in that local municipality per year and how many bottles of PDB one such infant requires for the first 14 days of life. From interactions with the SABR it was ascertained that an infant requires 20x250ml bottles of PDB for the first 14 days of life – this accounts for variability in the volumes contained in each bottle, waste due to thawed milk spoiling in the fridge and varying infant appetites [3, 4]. Data pertaining to the number of premature infants born without access to MoM simply do not exist. Conceivably, the Department of Health (DOH) should have data on the number of infants born per municipality per year as well as the percentage of these infants that classify as low-birthweight or premature infants, but this data could not be obtained. Population statistics combined with data from various local and international health reports were used to estimate the number of premature infants without access to MoM per local municipality in lieu of birth statistics.

To quantify supply, the question was: "How many bottles of UDB are supplied per local municipality per year?" This requires knowing the number of donors in a local municipality at any given time and the number of bottles expressed by a donor within a month. Interactions with the SABR confirmed that donors typically donate 20 bottles of UDB per month while they are active donors [3]. Determining the number of donors requires knowing how many HIV negative females are lactating within a local municipality at any given time and what percentage of these females would likely become regular donors while they are lactating. This data do not currently exist. Once again demographic data and health reports are used to estimate the number of HIV negative females lactating at a given time while literature regarding breastmilk banking is used to estimate the probability of these females becoming regular donors.

3 Determining demand of PDB

UNICEF reports that 1 092 000 infants were born in South Africa in 2007 [28], but this national figure cannot be disaggregated to local municipality level without much guesswork. Instead a "bottom-up" approach is followed. StatsSA reports the number of females per age group and race for each local municipality according to the 2001 census [21]. A study by the DOH reports the age distribution of pregnant women participating in an HIV screening survey from 2002-2004 [6]. The number of births per local municipality would thus depend partly on the number of women in each age band within that local municipality. According to the DOH, only 3% of births occur in women over 40. It was also assumed that minimal births would occur before the age of 15, thus only the age groups between 15 and 39 years were considered.

In another report from StatsSA [26], the estimated fertility rate (measured as number of live births per female in her lifetime) is given for each district municipality[4]. This rate was applied to the local municipalities within each district municipality. Within each local municipality, the fertility rate was adjusted according to the distribution of pregnancies across different age groups.

[4] A district municipality is a collection of local municipalities.

The expected births per local municipality are thus functions of the number of females within the local municipality and their age distributions as well as the fertility rate of the district municipalities the specific local municipalities fall under. Using this methodology and aggregating to a national level, the estimated number of annual births was 1 148 803 in 2001, 5.2% more than the UNICEF statistic for 2007 [28].

The latest national census was performed in 2011 and these results are not yet available. However, mid-year population estimates for 2011 were released by StatsSA [27]. Estimates were given for the female population per age group per province in 2011 showing a 10.83% overall growth in the female population between 15 and 39. Estimates of provincial fertility rates in 2011 are also given showing a decline of 0.29 in the national average. Adjusting these parameters in the demand model results in an estimate of the annual live births in 2011 of 1 147 445, 5.08% more than the UNICEF statistics for 2007 and 0.11% less than the estimate for 2001. The decline in births is attributed to the declining fertility rate and the fact that fertility rates are only available on a provincial level for 2011. When adjusting only the female statistics and not the fertility rate, annual births in 2011 are estimated at 1 277 193, 17% more than the UNICEF statistic for 2007 and 11.2% more than the 2001 estimate. The demand data is thus sensitive to the fertility rate, both in terms of accuracy and level of disaggregation.

Finding data regarding the percentage of premature or low-birthweight infants is trickier. It is estimated that in 1998 15% of infants born in South Africa were low-birthweight [28:6]. Little Steps [13] also cite the UNICEF report on low-birthweight and states that the estimated percentage of infants born prematurely in the public sector could be as high as 25%. This can probably be attributed to the fact that maternal health among the population that uses public health care is marginally lower than that of the population that have access to private health care. Due to the disparity in service quality between public and private health care in South Africa, it is assumed that if a household can afford to, they would opt for private care. Combining statistics regarding medical aid membership and household income levels [9], 2001 household income statistics per local municipality [22] and inflation statistics [25], the number of infants born in public care versus private care per local municipality was determined. Applying the assumption that 15% of the infants born in private care and 25% of the infants born in public care were low birthweight, the number of low-birthweight infants per local municipality can be determined.

What remains is to estimate what percentage of these low-birthweight infants does not have access to MoM. According to the SABR, the two most prevalent reasons premature infants do not have access to MoM in South Africa are poor maternal health (directly related to HIV/Aids and its concomitant opportunistic diseases) and the lack of rooming-in facilities in public hospitals. There is no data regarding the number of women that are affected by a lack of rooming-in facilities, but HIV prevalence among pregnant women is reported per province [6:8]. While some HIV positive mothers may still be able to lactate and chose to do so, the assumption is that if the mother was so ill that her infant was born prematurely, she is most likely also too ill to lactate. Because the number of HIV negative mothers affected by a lack of rooming-in facilities cannot be accounted for, the slight overestimation in assuming all HIV positive mothers of premature infants cannot lactate, is acceptable. Multiplying the number of premature infants per local municipality by the relevant provincial HIV prevalence statistic gives an estimate of the number of premature infants without access to MoM, which can then be multiplied by 20 to determine the bottles of PDB required by that local municipality per year.

Table 1 reports the national demand according to the number of premature infants requiring PDB per year and the resulting bottles of PDB required per year. A comparison is made between the results obtained from the 2001 statistics and those obtained when considering the 2011 population and fertility estimates [27].

Table 1: *National estimated demand for PDB in 2001 and 2011*

	Based on 2001 statistics	**Based on 2011 estimates**	**% change**
Premature infants requiring PDB per year	80 404	89 391	11.18%
Bottles of PDB required per year	1 608 080	1 787 822	

Figure 1 displays the concentration of demand on local municipality level. Hospitals with dedicated, sizable NICU facilities are also indicated.

4 Determining supply of UDB

UDB is sourced from lactating, HIV negative mothers who have undergone lifestyle screening and commit to periodic VCT testing. Studies have been done regarding the factors that influence a woman's choice to breastfeed and the duration of breastfeeding [2, 12, 14, 15]. The assumption is that women would no longer donate UDB if they are not breastfeeding their own infants. Ceriani Cernandas et al. [2] reports that the median duration of exclusive breastfeeding amongst a cohort of women is 4 months. The South Africa Demographic and Health Survey 2003 [5] shows that only 1.3% of infants are still exclusively breastfed at the age of 4 months. However, only 39.7% of 4 month old infants are not breastfed at all, the remainder are consuming liquids and/or complementary foods in addition to breastmilk. Women that are breastfeeding their infants in any capacity are included in the supply calculation, although it is more likely that a woman who is breastfeeding exclusively would donate as she is lactating in higher volumes and at shorter intervals.

Breastfeeding duration is also dependent on whether the mother has to return to work. In South Africa, double income households are the norm. Returning to work, whether part-time or full-time, usually initiates the process of weaning the infant from breastmilk. The assumption is that once the pressures of employment are reintroduced, a woman would no longer be willing to donate UDB, even if she still breastfeeds her child. In South Africa, maternity leave typically ranges between 3 and 4 months. Therefore, taking into account trends reported in literature and the average length of maternity leave, it is assumed that women would only donate UDB for 4 months after birth. The percentage of mothers breastfeeding (in any capacity) according to the age of their babies is 89%, 80%, 80% and 82% for 1 month, 2 month, 3 month and 4 month old babies, respectively [5].

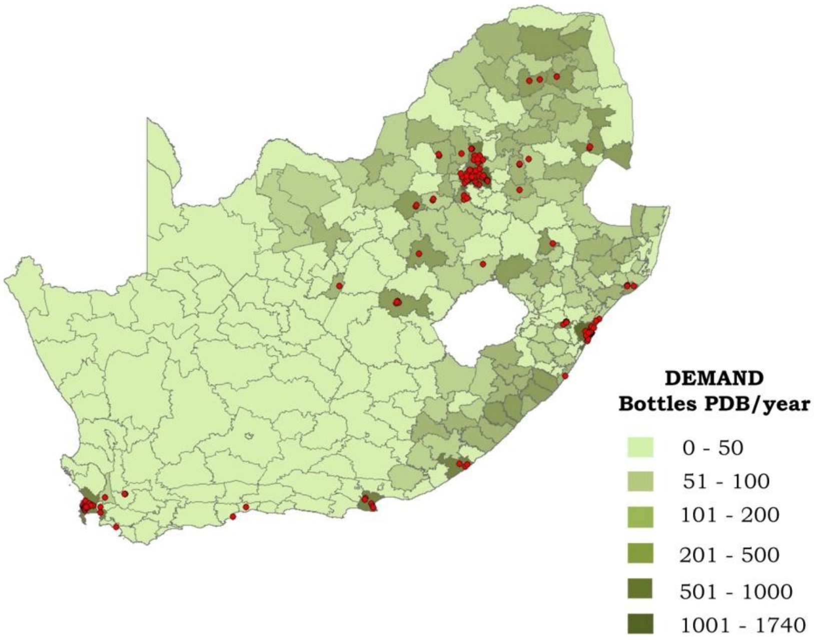


Figure 1: *Annual demand per local municipality with dedicated NICU facilities shown*

However, not all breastfeeding mothers would be willing to donate UDB. There have been studies regarding the demographic characteristics and motivations of breastmilk donors in France [1], the USA [16] and Brazil [17]. The findings from these studies are used to guide the assumptions regarding breastmilk donation in South Africa in lieu of South African studies. Regular donors are "*married, young, financially secure, well-educated, and healthy*" [16:355]. This assertion is supported by the work of Azema and Callahan [1] who state that donors are of average child-bearing age and relationally secure. Pimenteira Thomaz et al. [17] reports the average age of regular donors as 39.4 ffi 6.28 and that a higher educational level and the ability to stay at home influenced regular donation.

Women between the ages of 25-39 are considered likely donors in South Africa. To determine the number of women in this age group lactating at any given time, it is first calculated what the expected number of women (age 25-39) with infants 4 months and younger are per municipality and then breastfeeding percentages [5] are taken into account. Using the distribution of pregnancies across age groups [6], the number of expected births per annum in each municipality for women between the ages of 25 and 39 can easily be determined from the demand calculations.

Next the study has to account for lactating women in the age group of 25-39 years being "*married, young, financially secure, well-educated, and healthy*" [16:355]. An HIV negative mother is considered *healthy* and thus the HIV prevalence statistics among pregnant women [6] is used to only include HIV negative mothers. Mothers in stable relationships are more likely to donate [1, 16], likely due to an increased sense of physical, financial and emotional security. The 2001 South African Census [23] reports marital status of women in each local municipality. Being in any form of committed relationship (married or not) qualifies as *married* for the purposes of this study. To account for the *well-educated* criterion [16, 17] only the percentage of women with an education level of secondary school or higher is considered in each local municipality [24]. Lastly, household income is regarded as a proxy for *financial security*. Confined to the household income bands reported in StatsSA [22], household income of R76 800 in 2001 translates to R147 000 in 2011[5], which is still not enough to imply financial security. The next income band starts at R153 600 per annum in 2001, which translates to R295 000 in 2011 – sufficiently comfortable to denote financial security. The percentage of women considered financially secure in each local municipality is equal to the percentage of households earning more than R153 600 per annum in 2001. Due to South Africa's extreme economic inequality, the *financial security* percentages greatly diminish the pool of potential donors.

Undoubtedly there is correlation between the factors of health, wealth, education and relational stability of lactating females between the ages of 25-39. But in the absence of focused studies in this regard any assumptions would be no more than wild guesses. Having said that, education level and household income are generally highly correlated across all sectors of society and the household income criterion severely restricts the pool of potential donors. Erring on the conservative side, health and relational stability are applied independently of all the other variables. In the case of education and financial security a range of supply values is developed – the higher bound considering only the education level, the lower bound considering both education and financial security as independent variables and the intermediate value considering only financial security.

There are other, non-quantifiable, factors that influence breastmilk donation and its duration. These factors include ease of expressing breastmilk, awareness of the need for breastmilk banking, number of prenatal visits and previous pregnancies [1, 16, 17]. To account for these factors, it is assumed that only 40% of mothers who are currently breastfeeding infants younger than 4 months, are between the ages of 25 and 39, are HIV negative and are relationally secure, well-educated and financially secure would actually end up donating.

Table 2 reports the national supply according to potential donors at any given time and bottles of UDB per annum. A range of values shows the effect of the education level and financial security variables. A comparison is made between the results obtained from the 2001 statistics and those obtained when considering the 2011 population and fertility estimates [27]. Figure 2 shows the concentration of supply on local municipality level.

Table 2: *National supply of UDB in 2001 and 2011 for three different variable combinations*

	Based on 2001 statistics			**Based on 2011 statistics**			**% change**
Variables considered	Education level only	Financial security only	Financial security and education level	Education level only	Financial security only	Financial security and education level	Education level only
Potential donors at one time	3 013	634	225	3 171	673	239	+5.24%
Bottles per annum	723 091	152 149	53 970	760 997	161 697	57 448	

It is clear when comparing Figures 1 and 2 that there is significant geographic disparity between supply and demand which has implications for the network design. Supply is highly concentrated in the populous, economic centres of the country. Demand is also high in populous areas but is not bound to the economic centres (i.e. cities) and tends to be greater in areas with higher HIV prevalence. In addition, according to current supply assumptions, only 43% of the

[5] The adjustment uses the average inflation rate between 2001 and 2011 as calculated from the annual inflation on a monthly basis [25].

country's demand can be covered by available supply. Accounting for equitable distribution in the OR model will thus be imperative.

The sensitivity of the demand and supply data to the level of aggregation of the HIV prevalence statistic, breastfeeding assumptions, the percentage of newborns that are premature, and using either education level or financial security as a supply variable is explored in an interactive case study developed by the authors for teaching purposes.

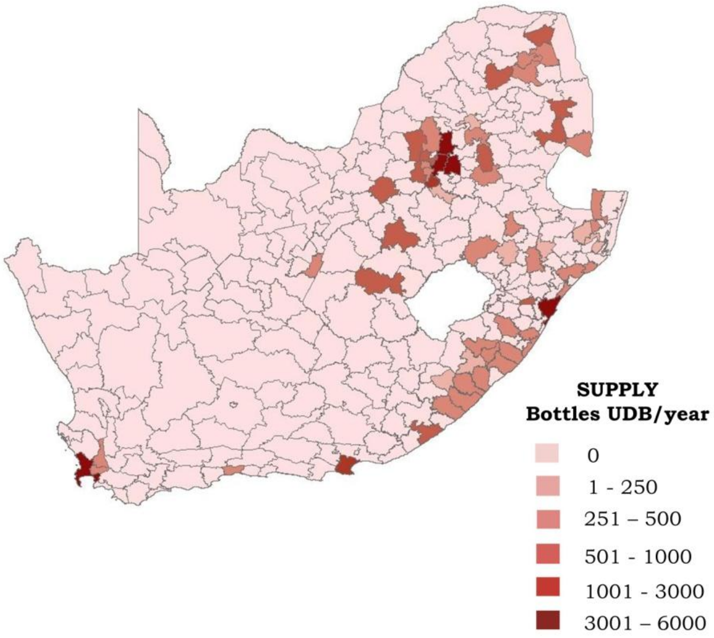


Figure 2: *Annual supply per local municipality*

5 Conclusion

National accessibility to donor breastmilk can protect thousands of premature infants born without access to MoM from fatal infections. The project team developed a facility location-allocation model to inform the national expansion of a breastmilk banking service model. This location-allocation model required disaggregate data on the number of premature infants born without access to MoM and the potential supply of UDB; data which are not currently available in South Africa. Through combining demographic data, selected health statistics and insights from literature and subject experts, a simple bottom-up methodology was developed for calculating demand and supply. Compared to available health statistics and an intuitive understanding of the country's demographics, these figures are valid enough to be used for the high-level, strategic planning of a national breastmilk donation network. The authors have identified three ways to further improve the currency and relevance of the demand and supply data namely: using the detailed 2011 census data when this becomes available, approaching the Department of Health and Department of Home Affairs for detailed birth statistics and conducting a South African breastmilk donor profile survey upon which donor assumptions could be based. These improvements are topics for future research.

Bibliography

[1] Azema, E., and Callahan, S. (2003). Breast milk donors in France: a portrait of the typical donor and the utility of milk banking in the French breastfeeding context. *Journal of Human Lactation*. 19(2): 199 – 202.

[2] Ceriani Cernandas, J.M., Noceda, G., Barrera, L., Martinez, A.M., and Garsd, A. (2003). Maternal and Perinatal Factors Influencing the Duration of Exclusive Breastfeeding During the First 6 Month of Life. *Journal of Human Lactation*. 19(2): 136-144.

[3] Cornelson, A. (2009). Personal interview via email. Conducted December 2009 by student interns at the CSIR, Pretoria.

[4] Dannheimer, W. (2009). *An inventory management system as a decision support tool for the South African Breastmilk Reserve*. Final year design project. University of Pretoria, Pretoria.

[5] Department of Health, Medical Research Council, OrcMacro. 2007. *South Africa Demographic and Health Survey 2003*. Pretoria: Department of Health. p142.

[6] Department of Health. (2004). *National HIV and Syphilis antenatal sero-prevalence survey in South Africa 2004*. Pretoria, South Africa.

[7] Farber C. (1998). HIV and breastfeeding: The fears. The misconceptions. The facts. *Mothering*.

[8] Harper, P.R., and Shahani, A.K. (2003). A decision support system for the care of HIV and Aids patients in India. *European Journal of Operational Research*. 147:187–197.

[9] Hospital Association of South Africa. (2009). *Private Hospital Review, 2009*. Hospital Association of South Africa. p 46.

[10] International Baby Food Action Network, Africa. (1999). IBFAN Africa statement on HIV and infant feeding. The IBFAN Africa Regional Workshop on Policy Guidelines for Infant Feeding and HIV, South Africa. August 1999.

[11] Jehan, I. et al (2009). Neonatal mortality, risk factors and causes: a prospective population-based cohort study in urban Pakistan. *Bulletin of the World Health Organization*. 87:130-138.

[12] Li, R., Fridinger, F., & Grummer-Strawn, L. (2002). Public Perceptions on Breastfeeding Constraints. *Journal of Human Lactation*. 18:227-235.

[13] Little Steps. Date unknown. What is Prematurity? Available: www.littlesteps.co.za/articles/what-is-prematurity. [Cited June, 2011].

[14] McLeod, D., Pullon, S., and Cookson, T. (2002). Factors Influencing Continuation of Breastfeeding in a Cohort of Women. *Journal of Human Lactation*. 18(4):335-343.

[15] O'Brien, M., Buikstra, E., Fallon, T., & Hegney, D. (2009). Exploring the influence of Psychological Factors on Breastfeeding Duration, Phase 1: Perceptions of Mothers and Clinicians. *Journal of Human Lactation*. 25:55-63.

[16] Osbaldiston, R., and Mingle, L. A. (2007). Characterization of Human Milk Donors. *Journal of Human Lactation*. 23(4): 350 – 357.

[17] Pimenteira Thomaz, A., Maia Loureiro, L.,V., Da Silva Oliviera, T., Furtado Montenegro, N., C., Almeida Junior, E.,D., Rodrigues Soriano, C.F., and Calado Cavalcante, J. (2008). The Human Milk Donation Experience: Motives, Influencing Factors, and Regular Donation. *Journal of Human Lactation*. 24(1): 69 – 76.

[18] Rahman, S., and Smith, D.K. (1990). Is 'appropriate OR' necessarily 'simple OR' for developing countries. *OPSEARCH*. 27(4):264-268.

[19] Rahman, S., and Smith, D.K. (2000). Use of location-allocation models in health service development planning in developing nations. *European Journal of Operational Research.* 123:437-452.

[20] South African Breastmilk Reserve (2011). SABR and Netcare: Neonatal Mortality Production Plan Report. Feed for Life Initiative: 2010/11, 2011/12.

[21] Statistics South Africa. (2006a). Table: Census 2001 by municipalities, age group, population group and gender. Statistics South Africa. Available: http://www.statssa.gov.za. [Cited February, 2011].

[22] Statistics South Africa. (2006b). Table: Census 2001 by municipalities, annual household income and population group of head of household. Statistics South Africa. Available: http://www.statssa.gov.za. [Cited February, 2011].

[23] Statistics South Africa. (2006c). Table: Census 2001 by municipalities, marital status, population group and gender. Statistics South Africa. Available: http://www.statssa.gov.za . [Cited February, 2011].

[24] Statistics South Africa. (2006d). Table: Census 2001 by municipality, highest level of education, population group and gender. Statistics South Africa. Available: http://www.statssa.gov.za. [Cited February, 2011].

[25] Statistics South Africa. (2009). *Annual inflation on a monthly basis: Consumer Price Index Statistical release: P0141*. Statistics South Africa, Pretoria.

[26] Statistics South Africa. (2010). *Estimation of fertility from the 2007 Community Survey of South Africa*. Statistics South Africa, Pretoria. p 18.

[27] Statistics South Africa. (2011). *Mid-year population estimates 2011. Statistical release: P0302*. Statistics South Africa, Pretoria.

[28] United Nations Children's Fund and World Health Organization. (2004). *Low birthweight: Country, regional and global estimates*. UNICEF, New York.

Proceedings of the 2012 ORSSA Annual Conference
pp. 88–98
www.orssa.org.za/wiki/uploads/Conf/2012ORSSAConferenceProceedings.pdf

ORSSA Proceedings
ISBN
978-07972-1406-4
© 2012

Results of Local-Search Heuristics for the Annual Crop Planning Problem

S Chetty[1] A Adewumi[2]

Abstract

This paper presents the results of three local-search (LS) metaheuristic algorithms for an NP-hard Annual Crop Planning (ACP) problem. A new LS technique called the Best Performance Algorithm (BPA) is introduced in literature. The results of BPA are compared against Tabu Search and Simulated Annealing in providing solution to this ACP problem.

Key words: Annual Crop Planning, Vaalharts Irrigation Scheme, Best Performance Algorithm

1 Introduction

Annual Crop Planning (ACP) is an NP-Hard [1] optimization problem in agricultural planning. It involves determining solutions that optimally allocate a limited amount of land, amongst the various competing crops that are required to be grown on it, within a year. Different types of crops start growing at different seasons, grow for different lengths of time and have different crop water requirements which are required for optimal growth. The objective of ACP is to determine solutions that seek to optimize the seasonal allocations of the limited amount of land, amongst the various competing crops, in a way that maximizes the total gross profits which can be earned while making the most efficient uses of the limited available resources. Determining 'optimal' solutions is necessary due to the increasing costs associated with agricultural production. The agricultural sector is also required to make more efficient use of fresh water supply. This is due to the increase in the demand for fresh water supply from other sectors such as domestic and industrial. There are multiple land and irrigation water allocation constraints associated with trying to determine optimal solutions to ACP.

Finding optimal solutions also involve taking into account the demands of the different types of crops grown, the soil characteristics, the climatic conditions, and other inputs associated with production such as the costs of labour and irrigated water supply [1]. To optimize irrigated water usage precipitation must be considered.

This paper introduces a new Annual Crop Planning (ACP) mathematical model, formulated by the authors in this paper. It is used to determine solutions concerning the optimal seasonal allocations of a limited amount of land, amongst the various competing crops which are required

[1] University of KwaZulu-Natal, South Africa, email: mervinthree@gmail.com

[2] Corresponding author: University of KwaZulu-Natal, South Africa, email: mervinthree@gmail.com, email: adewumia@ukzn.ac.za

to be grown on it, within a year. The objective aims to maximize the total gross profits which can be earned, while taking into account the variable costs associated with agricultural production.

Previous studies in crop and irrigation planning have used both single and multi-objective mathematical models. Many optimization techniques that have been used to provide solutions include Dynamic Programming (DP) [2], Genetic Algorithms (GA) [3], Evolutionary Algorithms (EA) [4] and Differential Evolution (DE) algorithms [5,6], amongst others. This paper also introduces a new local-search (LS) metaheuristic algorithm in literature. It is called the Best Performance Algorithm (BPA). The objective of this paper is to investigate and compare the performance of BPA against two other well-known LS metaheuristic algorithms, in providing solutions to an ACP problem at an existing irrigation scheme. These algorithms are Tabu Search (TS) and Simulated Annealing (SA).

2 The Annual Crop Planning Mathematical Model

This crop planning mathematical model is designed to maximize the total gross profits that can be earned from a given area of land, which has been allocated for agricultural production. The objective determine solutions that optimally allocates the land amongst the various competing crops which are required to be grown on it, within a year. Multiple land and irrigation water allocation constraints are considered.

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

Soil characteristics are also a factor in crop planning. Certain crops may adapt well only to certain types of soils. Therefore, the utilization of land is important for optimal yields. Irrigation application is also important. Too much or too little applications of water lead to sub-optimal plant growth. This will affect the yield of the crop. Soils are also sensitive to leaching due to excessive water applications [1]. Therefore, the seasonal irrigated water allocations need to be well planned.

A new Annual Crop Planning (ACP) mathematical model is introduced below. It takes into consideration the factors mentioned above and is formulated as follows:

2.1 Indices

- k – Plot types. (1 = single-crop plots, 2 = double-crop plots, 3 = triple-crop plots, and so on).
- i – Indicative of the sequential crops grown within the year, on plot type k (1 = 1st sequential crop, 2 = 2nd sequential crop, 3 = 3rd sequential crop, and so on).
- j – Indicative of the different types of crops grown at stage i, on plot type k.

2.2 Input Parameters

- l – Number of different plot types.
- N_k – Number of sequential crops grown within a year, on plot k.
- M_{ki} – Number of different types of crops grown at stage i, on plot k.
- L_{ki} – Total area of land allocated for agricultural production at stage i, on plot k.
- F_{kij} – Average fraction per hectare of crop j, at stage i, on plot k, which needs to be irrigated (1 = 100% coverage, 0 = 0% coverage).
- R_{kij} – Averaged rainfall estimates that fall during the growing months of crop j, at stage i, on plot k.
- CWR_{kij} – Crop water requirements of crop j, at stage i, on plot k.

- T – Total hectarage of land allocated for agricultural production for the year.
- A – Volume of irrigated water that can be supplied per hectare (ha^{-1}).
- P – Price of irrigated water per m^3.
- O_{kij} – Other operational costs ha^{-1} of crop j, at stage i, on plot k. This cost excludes the cost of irrigation.
- YR_{kij} – The amount of yield obtained in tons per hectare (t ha^{-1}) of crop j, at stage i, on plot k.
- MP_{kij} – Producer prices per ton of crop j, at stage i, on plot k.
- Lb_{kij} – Lower bounds of crop j, at stage i, on plot k.
- Ub_{kij} – Upper bounds of crop j, at stage i, on plot k.

2.3 Calculated Parameters

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

2.4 Variables

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

2.5 Objective Function

Maximize $f =$

$$\sum_{k=1}^{l}\sum_{i=1}^{N_k}\sum_{j=1}^{M_{ki}} X_{kij}B_{kij} \qquad (1)$$

This objective is subjected to the land and irrigated water allocation constraints given below.

2.6 Land Constraints

The sum of the amount of land allocated for each crop j, at stage i, on plot k, must be less than or equal to the total area of land allocated for agricultural production at stage i, on plot k.

$$\sum_{j}^{M_{ki}} X_{kij} \leq L_{ki} \quad \forall k, i \qquad (2)$$

Feasible solutions must satisfy lower and upper bound constraints. This will ensure that the near-optimal solution found will be relative to the demands of the current agricultural practices.

$$Lb_{kij} \leq X_{kij} \leq Ub_{kij} \quad \forall k, i, j \quad (3)$$

2.7 Irrigation Constraints

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

$$\sum_{k}\sum_{i}\sum_{j} IR_{kij} \leq TA \qquad (4)$$

3 Case study

Situated on the border, separating the Northern Cape from the North West Province in South Africa, lays the Vaalharts Irrigation Scheme (VIS). The VIS is one of the largest irrigation schemes in the world. It comprises approximately of 36 950 ha of agricultural land [7]. The area is known for its cold frosty winters and warm summers [8]. However, despite the climate this scheme produces some of the best agricultural produce in the country.

The average annual rainfall in this area, determined over a period of 36 years, is slightly under 450 mm. 89% of this rainfall falls between the months of October to April [8]. For prime agricultural production at the VIS, it is necessary that irrigated water be used to supplement the lack of rainfall in the area.

Irrigated water for the VIS is drawn from the Vaal River. This water is transported by two main canals, the North canal and the West canal. Each canal supplies a network of canals which in turn supply the feeder canals. The feeder canals supply irrigated water to the farm plots [7]. A maximum volume of 9140 m^3 ha^{-1} annum^{-1} of irrigated water is supplied to the farmers. A water charge of 8.77 cents/m^3 needs to be paid to the Vaalharts Water User Association [7].

The statistics of the primary crops grown in this area is given in Table 1 [8]. These statistics have been determined over a 5 year period. It includes the hectares planted per crop (ha's crop^{-1}), and the average tons of returns produced per hectare per crop (t ha^{-1}). The crops cultivated consists of yearly (y) and perennial (p) crops. These are grown on single-crop plots. The rest consist of summer (s) and winter (w) crops. These are grown on double-crop plots and on triple-crop plots.

At current practices, the total amount of land allocated for the yearly and perennial crops is calculated to be 8300 ha. The total amount of land allocated for the summer crops is 15500 ha. The land allocated for the winter crops is 12200 ha.

The Crop Water Requirements (CWR) for each crop is provided by [9]. The average rainfall (AR) for the months that each crop grows is determined from [8]. The producer prices for a ton of yield produced from each crop (ZAR t^{-1}) is given by [10].

Table 1: *Vaalharts irrigation scheme crop statistics*

Crop Types	ha's crop-1	t ha^{-1}	CWR	AR	ZAR t^{-1}
Pecan nuts (y)	100	5.0	1600	444.7	3 500.00
Wine Grapes (y)	300	9.5	850	350.8	2 010.00
Olives (y)	400	6.0	1200	444.7	2 500.00
Lucerne (p)	7500	16.0	1445	444.7	1 185.52
Cotton (s)	2000	3.5	700	386.4	4 500.00

Maize (s)	6500	9.0	979	279.0	1 321.25
Groundnuts(s)	7000	3.0	912	339.5	5 076.00
Barley (w)	200	6.0	530	58.3	2 083.27
Wheat (w)	12000	6.0	650	58.3	2 174.64

4 Methodology

LS metaheuristic algorithms have provided effective solutions to many real-world NP-Hard optimization problems. This paper introduces a new LS metaheuristic algorithm in literature. It is formulated by the authors of this paper and is called the Best Performance Algorithm (BPA). The performance of this algorithm is compared against the performances of TS and SA in them providing solutions to the ACP problem at the VIS. Brief descriptions of BPA, TS and SA are given below.

4.1 Best Performance Algorithm

The Best Performance Algorithm is modelled on the competitive nature of professional athletes, in them desiring to perform at their best within competitive environments. To perform at their best, these athletes would need to strategize and practice. Strategizing and practice will help them improve their talents by developing refined skills. These refined skills will enable the athletes to push the boundaries of their best performances, irrespective of their sporting disciplines.

An effective strategy used in trying to improve on performances is to make use of technology. Technology can be used to identify the athletes' strengths and weaknesses in them delivering a performance. By improving on weaknesses or even developing new techniques the athlete might be able to register improved performances. One way to use technology is to maintain an archive/collection of the athletes best registered performances. This collection will provide the athlete with a reference to go back to in order to review the technique(s) used in delivering a previous best performance. Upon reviewing a previous best performance the athlete may be able to make appropriate changes, in the hope of trying to determine improved technique(s). Improved techniques will lead to improved performances being delivered within competitive environments. BPA is modelled on the idea of an athlete trying to develop refined skills, for the sake of improved performances, by maintaining a collection of his/her best registered performances.

BPA is implemented by maintaining a *sorted* list of a limited number of the individual athletes' best performances. This list is called the Performance List (PL). The better the quality of a performance the higher up on the list will be its ranking. In trying to develop refined skills or possibly determining a new technique the athlete will review a performance from the PL, and will then seek to make appropriate changes. By making slight changes (performing LS) improved technique(s) may possibly be determined. If an improved technique is found, which leads to an improved performance being delivered, then the PL will be updated with this performance. This improved performance *must* at least improve on the *worst* performance registered on the PL. The worst performance will get removed when the PL is updated with an improved performance. The sorted order of the PL must always be maintained. No identical performances are allowed in the PL. The athlete may continue to work with the technique(s) previously worked with or choose another performance to work with from the PL. After a sufficient amount of strategizing and practice the athlete would have determined his/her best technique to use.

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

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

The algorithm for the BPA is as follows:

```
1. Set the index variable, $index = 0$
2. Set the size of the Performance List, $listSize$
3. Initialize probability, $p_a$
4. Populate the Performance List $(PL)$ with random solutions
5. Calculate the fitness values of the solutions in $PL$, i.e. $(PL\_Fitness)$
6. Sort $PL$ and $PL\_Fitness$ according to $PL\_Fitness$
7. Initialize $working$ to $PL_{index}$
8. for $i$ to $noOfIterations$ do
    8.1. $working$ = Perform˙LS($working$)
    8.2. $f\_working$ = Evaluate ($working$)
    8.3. if $f\_working$ better then $PL\_Fitness_{listSize-1}$ then
        8.3.1. Update $PL$ with $working$
        8.3.2. Update $PL\_Fitness$ with $f\_working$
    8.4. end if
    8.5. if random[0,1] ¿ $p_a$then
        8.4.1. $index$ = Random[0,$listSize$]
        8.4.2. $working = PL_{index}$
    8.6. end if
9. end for
10. return $PL_0$
```

4.2 Tabu Search

Tabu Search (TS) [13,14] is based on the idea of something that should not be interfered with. TS implements this idea by recording a specific number of unique best solutions found in a list called the Tabu List (TL). If a new solution is found, which improves on the solutions recorded in the TL, the new solution gets added to the TL. Any new solutions found that is identical to those that are already registered in the TL will not considered. This eliminates the possibility of exploiting identical moves.

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

The algorithm for TS is as follows:

```
1. Generate an initial random solution = $best$
```

```
2. Set $current = best$
3. Evaluate the fitness of $best = f\_best$
4. Set the fitness of $current$ $(f\_current) = f\_best$
5. Initiate the Tabu List $TL$ and the $CandidateList$
6. for $i$ to $noOfIterations$ do
   6.1. $CandidateList$ = Generate˙List($current$)
   6.2. $current$ = Find˙Best˙Candidate($CandidateList$)
   6.3. $f\_current$ = Evaluate˙Fitness($current$)
   6.4. if $f\_current$ better then $f\_best$ then
        6.4.1. $f\_best = f\_current$
        6.4.2. $best = current$
        6.4.3. Update $TL$ with $current$
   6.5. end if
7. end for
8. return $best$
```

4.3 Simulated Annealing

Simulated Annealing (SA) [15] models the annealing process, when heated metal begins to cool. The hotter metal gets, when heated, the more volatile it atomic structure will become. This will result in a weakened and more unstable structure. However, when the heated metal begins to cool the highly energized metallic atoms will lose energy and the structure will begin to stabilize. When the metal is completely cooled, an equilibrium state is reached. The cooling process must be slow for the annealing to be successful.

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

The algorithm for SA is as follows:

```
1. Generate an initial random solution = $best$
2. Set $current = working = best$
3. Evaluate the fitness of $best = f\_best$
4. Set the fitness of $current$ $(f\_current)$ and the fitness of $working$ $(f\_working) = f\_best$
5. Initiate starting temperature $T$ and final temperature $F$
6. while $T \geq F$ do
    6.1. for $i$ to $stepsPerChange$ do
         6.1.1. $working$ = Generate˙Solution($current$)
         6.1.2. $f\_working$ = Evaluate˙Fitness($working$)
         6.1.3. if $f\_working$ better then $f\_current$ then
                6.1.3.1. $use\_solution$ = true
         6.1.4. else
                6.1.4.1. Calculate acceptance probability $P$
                6.1.4.2. if $P >$ random[0,1] then
                         6.1.4.2.1. $use\_solution$ = true
               6.1.4.3. end if
```

```
            6.1.5. end else
            6.1.6. if $use\_solution$ then
                6.1.6.1. $use\_solution$ = false
                6.1.6.2. $f\_current = f\_working$
                6.1.6.3. $current = working$
                6.1.6.4. if $f\_current$ better then $f\_best$ then
                    6.1.6.4.1. $best = current$
                    6.1.6.4.2. $f\_best = f\_current$
                6.1.6.5. end if
            6.1.7. end if
        6.2. end for
        6.3. Update $T = (T \times \alpha)$
7. end while
8. return $best$
```

5 Experiment and Results

The program was written in Java, using the Netbeans 7.0 IDE. The program was run on the same platform using a computer with an Intelff Pentium™ 4 processor and 3 GB of RAM.

The non-heuristic specific parameters, required for the execution of the algorithms, had been set according to the values given in Table 2. The lower and upper bounds ensure that feasible solutions are found which relates to the current agricultural practices at the irrigation scheme. $F_{kij} \in [0,1]$. C_IR_{kij} is the cost of irrigated water (ZAR ha^{-1}). O_{kij} is set to a third of the producer prices per ton of yield (ZAR ha^{-1}). These values are sufficient to evaluate the performances of the heuristic algorithms, in comparing them to the results of the current agricultural practices.

Table 2: *Non-heuristic specific parameters required for the execution of the algorithms*

Crop Types	Lb_{kij}	Ub_{kij}	F_{kij}	C_IR_{kij}	O_{kij}
Pecan nuts	30	200	1	1 013.20	5 833.35
Wine Grapes	100	500	1	437.80	6 365.00
Olives	150	600	1	662.40	4 999.98
Lucerne	7000	8000	1	877.26	6 322.72
Cotton	1000	3000	1	275.03	5 250.00
Maize	4000	8000	1	613.90	3 963.78
Groundnuts	4500	9500	1	502.08	5 076.00
Barley	50	350	1	413.68	4 166.52
Wheat	11850	12150	1	518.92	4 349.28

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

- BPA – $noOfIterations$ was set at 20000. $listSize$ was set at 20. p_a was set at 0.2.
- TS – $noOfIterations$ was set at 1000. The $CandidateList$ size was set at 20. The TL size was set at 7.
- SA – The initial temperature T was set at 50. The final temperature F was set at 0.9. α was set at 0.99. The $stepsPerChange$ set at 50.

These parameter settings ensure that each algorithm execute 20 000 objective function evaluations. Each algorithm was also run 100 times. Graphical representations of the results found are shown in Figures 1, 2 and 3. Figure 1 shows the average execution times of each algorithm. It is seen that the execution times are all comparable. Figure 2 shows the averaged and best fitness values found. These values are determined from Table 3 below. BPA marginally provides the best solutions, both on average and at best, compared to TS. The solutions of SA are inferior. Figure 3 shows the hectares allocated per crop by each metaheuristic algorithm and also that of the current agricultural practices (CP) at the scheme. It is seen that each algorithms determine similar hectare allocations per crop type.

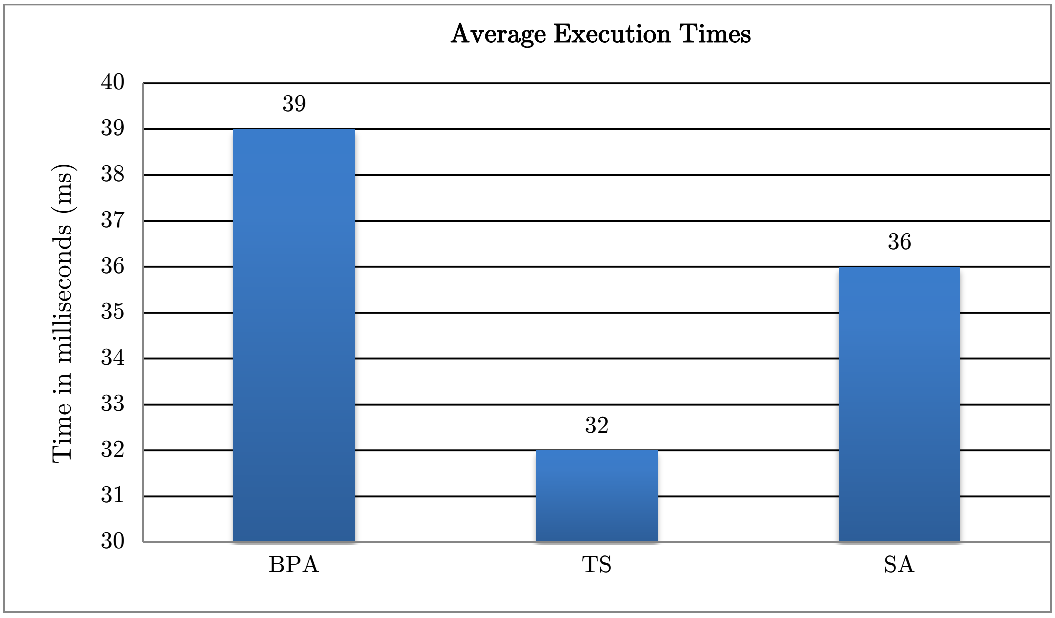


Figure 1: *The average execution times in milliseconds*

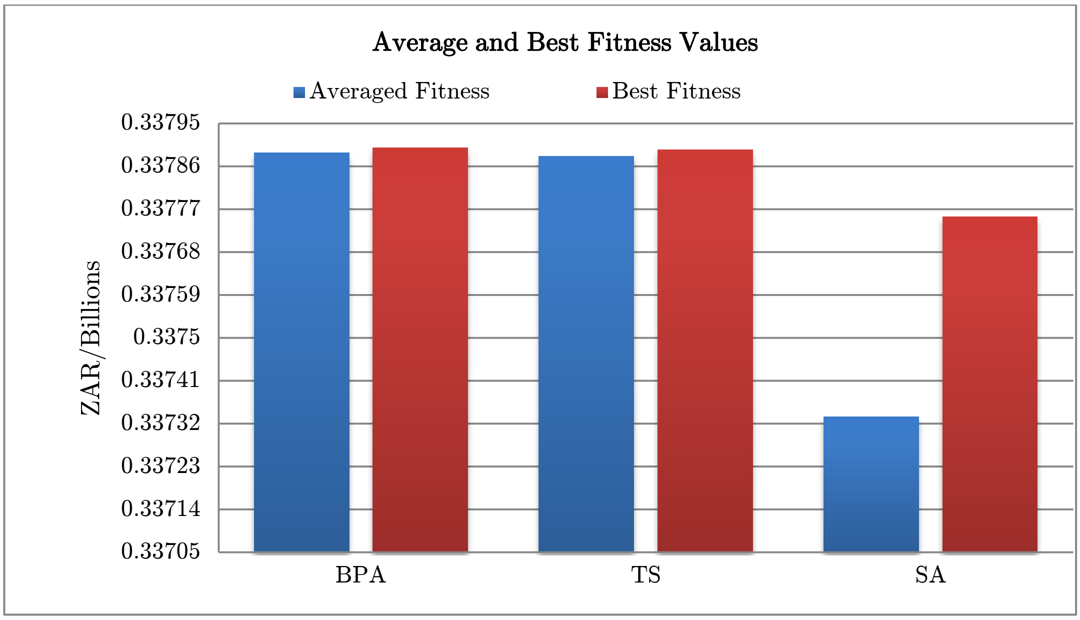


Figure 2: *The average best fitness and overall best fitness values*

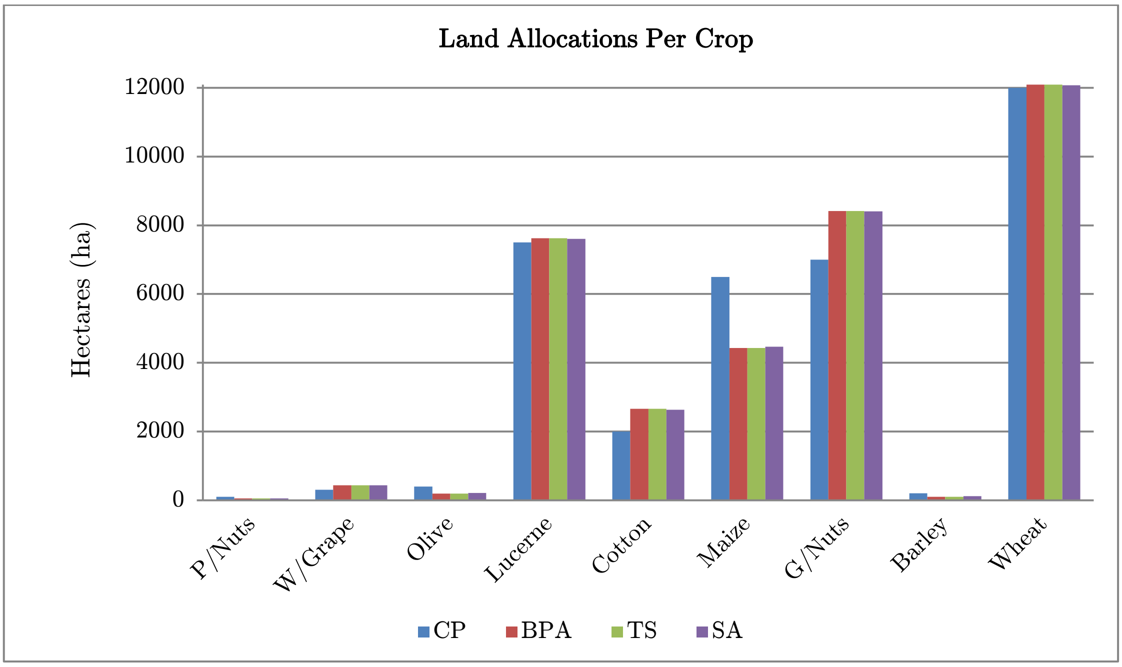


Figure 3: *Comparison of land allocations per crop*

Table 3 below gives the statistical values of the average (AFV) and best (BFV) fitness values found. It also gives the values of the Irrigation Water Requirements (IWR) and the Variable Costs of Production (VCP), for each of the best heuristic solutions found. This is compared against the statistics of the CP. It is observed that each metaheuristic algorithm determines improved solutions compared to CP. Each heuristic solution shows increased gross profits (BFV) and decreased volumes of irrigated water requirements (IWR). BPA saves a total volume of 4 467 915 m^3, TS a total of 4 449 185 m^3 and SA a total of 4 421 467 m^3 of irrigated water. At the quota of 9 140 m^{3}ha^{-1}annum^{-1} the savings determined by BPA, TS and SA would be able to supply irrigated water for another 488 ha, 486 ha and 483 ha of agricultural land, respectively.

Table 3: *Statistics of the solutions determined by each heuristic algorithm*

HA	AFV (ZAR)	BFV (ZAR)	IWR (m^3)	VCP (ZAR)
CP	N/A	332027707	244491000	198176322
BPA	337888406	337899527	240023085	200524382
TS	337881636	337894726	240041815	200524445
SA	337334688	337754412	240069533	200457937

The strength of BPA, in determining the overall best solutions, is attributed to its ability to maintain an *updated* collection of its best solutions found. Each improved solution represents a more promising area found within the local neighbourhood structures of the solution space. Maintaining a collection of the best solutions found encourages exploration of the different neighbourhood structures. The probability p_a encourages exploitation within those local neighbourhood structures. TS's strong exploitation ability enables it to perform better, compared to SA, for this type of problem.

5 Conclusion

The paper presents comparative results of three local-search metaheuristic algorithms for an NP-hard ACP problem at the Vaalharts Irrigation Scheme (VIS). Each heuristic algorithm determines improved solutions, compared to the current practices at the VIS. In a solution space where the dimensions remain constant our new heuristic algorithm, BPA, marginally provides the best overall solutions.

Bibliography

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

[2] Sunantara, J. D. and Ramirez, J. A. (1997). Optimal stochastic multicrop seasonal and intraseasonal irrigation control. *Journal of Water Resources Planning and Mangement*, 123(1): 39-48.

[3] Wardlaw, R. and Bhaktikul, K. (2007). Application of genetic algorithms for irrigation water scheduling. *Irrigation and Drainage*, 53:397–414.

[4] Sarker, R. and Ray, T. (2009). An improved evolutionary algorithm for solving multi-objective crop planning models. *Computers and Electronics in Agriculture*. 68 (2):191–199.

[5] Adeyemo, J., Bux, F. and Otieno, F. (2010). Differential evolution algorithm for crop planning: single and multi-objective optimization model," *International Journal of the Physical Sciences,* 5(10):1592-1599.

[6] Adeyemo, J. and Otieno, F. (2010). Maximum irrigation benefit using multi-objective differential evolution algorithm (MDEA). International Journal of Sustainable Development, 1: 39-44.

[7] Grove, B. (2008). Stochastic Efficiency Optimisation Analysis of Alternative Agricultural Water Use Strategies in Vaalharts over the Long- and Short-Run. Ph.D. thesis. Department of Agricultural Economics, Univ. of the Free State, Bloemfontein, South Africa.

[8] Maisela, R. J. (2007). Realizing agricultural potential in land reform: the case of Vaalharts Irrigation Scheme in the Northern Cape Province, M. Phil Mini-Thesis, University of the Western Cape, South Africa.

[9] Irrigation Water Management: Irrigation Water Needs. [Online]. Available: http://www.fao.org/docrep/S2022E/S2022E00.htm#Contents

[10] Department of Agriculture, Forestry and Fisheries (2012). Abstract of Agricultural Statistics [Online]. Available: http://www.nda.agric.za/docs/statsinfo/Ab2012.pdf

[11] Sorensen, K. and Glover, F. (2010). Metaheuristics. In Gass, S.I. and M.C. Fu (ed), *Encyclopedia of Operations Research and Management Science,* 3e, Springer, New York.

[12] Jain, K. S. and Singh, P. V. (2003). Water Resource Systems Planning and Management. *Development in Water Science, 51:3-858,* Elsevier.

[13] Glover. F. (1989). Tabu Search - Part 1. *ORSA Journal on Computing* 1 (2): 190–206.

[14] Glover. F. (1990). Tabu Search - Part 2. *ORSA Journal on Computing* 2 (1): 4–32.

[15] Tan, C.M. (2008). Simulated Annealing. In-Teh, ISBN-13: 978-953-7619-07-7.

Proceedings of the 2012 ORSSA Annual Conference
pp. 99–106
www.orssa.org.za/wiki/uploads/Conf/2012ORSSAConferenceProceedings.pdf

ORSSA Proceedings
ISBN
978-07972-1406-4
©2012

The Potential of Self-Organising Traffic Control Paradigms

MD Einhorn* AP Burger† JH van Vuuren‡

Abstract

In this paper, a traffic control technique is presented which utilises self-organising algorithms to determine the switching sequences of traffic control signals at an intersection. The data utilised by the algorithms are provided by radar detection equipment mounted at the intersection, allowing the traffic lights to monitor a length of roadway and determine the most effective switching strategy based on the speeds of the approaching vehicles as well as their respective distances from the intersection. The self-organising algorithms investigated are tested in a simulated environment using a computerised traffic simulation model designed and built specifically for the purpose of the study. The results are compared to those obtained by a fixed-time regime for which the green times have been optimised, in terms of average time spent by vehicles in the system and the average queue lengths for the system.

Key words: Traffic control, self-organisation, simulation.

1 Introduction

With an ever increasing traffic demand along the roadways of urban cities around the world comes an ever increasing need for improved traffic regulation and control [1]. Certain infrastructures are in place in various cities around the world which attempt to alleviate traffic congestion as far as possible and thus the negative social, economic and environmental effects associated with it. In terms of traffic control at intersections, various techniques have been implemented to better facilitate traffic flow along arterials through the introduction of optimisation techniques which seek to implement the most appropriate green times, cycle lengths and offsets between adjacent intersections for different times of the day and their associated traffic flows [4].

*Corresponding author: Department of Logistics, Stellenbosch University, Private Bag X1, Matieland, 7602, South Africa, email: 14854937@sun.ac.za

†Department of Logistics, Stellenbosch University, Private Bag X1, Matieland, 7602, South Africa, email: apburger@sun.ac.za

‡Department of Logistics, Stellenbosch University, Private Bag X1, Matieland, 7602, South Africa, email: vuuren@sun.ac.za

Recent improvements in traffic monitoring technology have opened up new opportunities for improved traffic control. One such example is the introduction of radar detection technology [6] utilised by centralised traffic management control centres to provide information pertaining to vehicle flow rates and incident detection so that controllers can use the information provided to make the necessary adjustments to the traffic control techniques in place. However, the technology is also capable of detecting and tracking individual vehicles, providing information such as the vehicle's speed, distance from a particular point, distance to the vehicle in front of it, among others.

The objective in this paper is to determine how this new and relevant technology may be used to improve traffic flow at intersections in terms of reducing the waiting times of vehicles in the network. The data provided by the radar technology are used as input to two *self-organising* traffic control algorithms which offer a decentralised approach to traffic control in a network. The effectiveness of these algorithms in alleviating congestion and reducing waiting times is investigated in a simulated environment.

Self-organisation may be applied to the control of traffic at an intersection if such an intersection is fitted with the aforementioned radar detection equipment, effectively allowing the traffic lights to observe the current traffic situations along each of the roadways approaching the intersection. The crux of such an application is to develop algorithms which make use of the data supplied by the radar detection equipment in such a way that they locally utilise intersection capacity as efficiently as possible so as to effectively reduce network-wide congestion and driver waiting times.

The approach of self-organising traffic light control differs from that of conventional adaptive traffic light control in that instead of attempting to optimise the green splits, cycle times and offsets of a number of adjacent intersections in a coordinated fashion, effectively coercing traffic to adjust to the signal timings, self-organising traffic light control decentralises the control in the network such that each intersection is responsible for employing the best signal timings relative to the current traffic approaching it, thus ensuring that the signals adapt to the approaching traffic and not the other way around. A consequence of each intersection in a traffic network being optimised locally is that a global ripple-effect occurs, resulting in a natural system-wide traffic signal synchronisation among intersections as opposed to the co-ordinated synchronisation attempted by global optimisation techniques.

2 Self-organisation

Self-organisation is an optimisation technique inspired by numerous processes that occur in nature in which the system regulates itself in response to external environmental factors. This regulation, however, is not implemented through a central point of control, but instead, through numerous local interactions which are governed by an often simple set of local rules, and it is from these local interactions that a global optimum is achieved.

A formal definition of a self-organising system is provided by Serugundo *et al.* [5] as follows:

> "A self-organising system functions without central control, and through contextual local interactions. Components achieve a simple task individually, but a

complex collective behaviour emerges from their mutual interactions. Such a system modifies its structure and functionality to adapt to changes to requirements and to the environment based on previous experience."

3 Designing a self-organising traffic control algorithm

An excellent example of a self-organising traffic control algorithm was presented by Lämmer and Helbing in [2]. Their approach to self-organising traffic control was inspired by observing the change in flow direction of pedestrians through a doorway due to "pressure" differences. In particular, they noticed that as the number of pedestrians waiting on one side of the doorway increased, so did the pressure they exerted on it, until this pressure exceeded that on the other side of the doorway by a sufficient amount, at which point the passing direction of people through the doorway would change. This notion of pressure was translated to traffic control by means of pressure indices. Lämmer and Helbing determined these pressure indices for each traffic flow approaching an intersection based on predetermined arrival and departure rates, as well as anticipated green times and resulting waiting times. More specifically, the pressure index for traffic flow i approaching an intersection at time t is given by

$$\pi_i^I(t) = \frac{\hat{n}_i(t)}{\tau_{i,\sigma}^{\text{pen}}(t) + \tau_i(t) + \hat{g}_i(t)}, \tag{1}$$

which may be interpreted as a measure of the anticipated average service rate, or more accurately, the anticipated number of vehicles expected to receive service, $\hat{n}_i(t)$, during a time period of length $\tau_{i,\sigma}^{\text{pen}}(t) + \tau_i(t) + \hat{g}_i(t)$. This definition depends on the anticipation of vehicle arrivals and the anticipated green time, $\hat{g}_i(t)$, required to clear the intersection, and takes into account the time losses associated with switching service from one traffic flow to another (typically the time of the amber and all red signal periods during which the intersection may not be utilised by any traffic flow, given by $\tau_i(t)$) as well as switching service back at a later stage (given by the penalty term $\tau_{i,\sigma}^{\text{pen}}(t)$).

As a precautionary measure, to ensure that vehicle queues do not grow excessively large or that certain traffic flows do not receive a green signal for unreasonably long periods of time, Lämmer and Helbing introduced two stabilising, user-defined parameters, T and T^{max}, with $T \leq T^{max}$, such that each traffic flow approaching an intersection was served once on average every T seconds, and at least once every T^{max} seconds. Together with anticipated vehicle arrival and departure rates, as well as anticipated green times, the parameters T and T^{max} were used to determine maximum allowable green times and queue lengths algebraically (*i.e.* based on averaged vehicle flow rates only, without using the kinematic data of individual vehicles).

We expect certain aspects of the approach of Lämmer and Helbing to provide a suitable platform around which alternative self-organising traffic control algorithms may be built which utilise the kinematic data of individual vehicles. In particular, investigating alternatives to the pressure index expression in (1) is certainly desirable. It is felt that a greater degree of accuracy may be associated with the actual vehicle arrival data provided by the radar detection equipment as opposed to the approximated vehicle arrival rates employed by Lämmer and Helbing, which may be used to develop more accurate, real-time pressure indices. A self-organising traffic control algorithm should be simple and contain as few user defined parameters as possible, instead relying more on the data generated by the local, real-time traffic conditions, as relayed by the radar detection equipment, to determine effective and efficient signal switching schedules. A visual representation of some of the data provided by the radar detection equipment may be found in Figure 1.

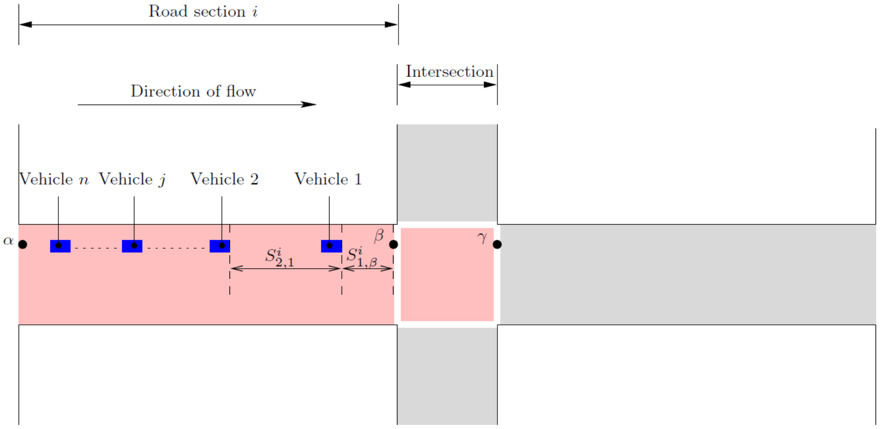


Figure 1: Example of a road section.

A road section approaching an intersection is shown in Figure 1. The stopping point of the road section at the intersection is labelled β. For a given number of vehicles, n, within the detection range of the radar detection equipment, information may be provided for each individual vehicle j along road section i with respect to its speed, v^i_j, its distance to the stopping point of road section i, $S^i_{j,\beta}$, and its distance to the vehicle in front of it, $S^i_{j,j-1}$, among others.

Since the radar detection equipment provides information on both the speed and the distance to the intersection of each vehicle along a roadway, it is possible to calculate the time it will take each vehicle on road section i to reach the intersection by dividing this distance by its speed. With this in mind, the simple expression

$$\sum_{j=1}^{n} \frac{S^i_{j,\beta}}{v^i_j}. \tag{2}$$

may be used when determining which traffic flow should receive service. The expression in (2) above may be interpreted as the sum of the times required for each vehicle along road section i to reach the intersection. The reasoning and logic behind (2) is as follows: The closer a vehicle is to the intersection, and the faster the speed at which it is travelling, the more urgently it requires service compared to a vehicle that may be further from the intersection and/or travelling at a slower speed. This increased urgency for service coincides with a shorter time to arrival at the intersection. Thus, when summing together the times to arrival at the intersection of all vehicles along each road section, it may be inferred that the vehicles along the road section with the minimum sum would require service most urgently.

With this logic in mind, it was decided to incorporate the time losses incurred when servicing each traffic flow in an attempt to effectively account for queued, stationary vehicles as well. The notion of overall time loss may be used to decide which traffic flow should receive service. More specifically, the priority value of traffic flow i may be approximated by the waiting time that will be experienced by all other traffic flows approaching the intersection, coupled with the waiting time they have experienced up to the current time if traffic flow i were to receive service, *i.e.* if $i = \sigma$, where σ indicates the index of the traffic flow currently receiving service. In particular, the priority index of traffic flow i may be taken as

$$\pi_i^{II}(t) = \begin{cases} \overbrace{\sum_{k \neq i}\sum_{j=1}^{m} w_j^k(t)}^{a} + \overbrace{\sum_{j=1}^{n} \frac{S^i_{j,\beta}(t)}{v^i_j(t)}}^{b} + \overbrace{A + R}^{c} & \text{if } i = \sigma \\ \underbrace{\sum_{j=1}^{m} \sqrt{\frac{2S^i_{j,\beta}(t)}{a^i_j}}}_{d} + \underbrace{\sum_{j=m+1}^{n} \frac{S^i_{j,\beta}(t)}{v^i_j(t)}}_{e} & \text{if } i \neq \sigma. \end{cases} \tag{3}$$

For the case in which traffic flow i is receiving service at time t (*i.e.* $i = \sigma$), the first term on the right hand side of (3), labelled a, is the sum of all waiting times experienced by currently queued vehicles not in traffic flow i up to time t. The second term, labelled b, is the sum of the times to arrival at the intersection of all vehicles along traffic flow i in (3) and is an approximation of the additional time that currently queued vehicles not in traffic flow i will have to continue waiting for service while all detected vehicles along traffic flow i are cleared. The third term, labelled c, is the sum of the amber time A and the all-red time R during which all currently queued vehicles will have to wait through once it has been decided to switch service.

For the case in which traffic flow i is not currently receiving service (*i.e.* $i \neq \sigma$), the first term on the right hand side of (3), labelled d, is an approximation of the time it will take to clear the m currently queued vehicles, which would start accelerating from rest with an acceleration of a_j. The second term, labelled e, is an approximation of the time it would take to clear all other vehicles approaching the intersection along traffic flow i that are not yet queued at time t.

The priority index $\pi_i^{II}(t)$ may therefore be interpreted as follows: if traffic flow i is currently receiving service, $\pi_i^{II}(t)$ is the sum of the times vehicles not in traffic flow i have spent waiting up to time t together with an approximation of how much longer they are expected to remain waiting while traffic flow i continues to receive service. For the case in which traffic flow i is not receiving service, $\pi_i^{II}(t)$ may be interpreted as the waiting time that will be experienced by all vehicles not in traffic flow i if traffic flow i does, in fact, receive service.

Service is awarded to the traffic flow achieving the smallest priority index as this represents the minimum overall waiting time that vehicles in the system have to incur. That is, a traffic flow will continue receiving service until the waiting time experienced by all other vehicles not receiving service together with predicted future waiting times, and amber and all-red times, exceeds that which would be experienced if service were switched to an alternative traffic flow.

When a switch in service is made to traffic flow i, it receives a green signal for at least as long as the value of $\pi_i^{II}(t)$ when the switch in service was made (as this represents the amount of time required to clear the currently waiting queue as well as all other detected vehicles approaching the intersection along traffic flow i), and no longer than a certain maximum allowable green time, $G^{\max}$. A time $\min\{\pi_i^{II}(t), G^{\max}\}$, after the commencement of service to traffic flow i, if the priority index of traffic flow i is still a minimum, then traffic flow i will continue to receive service, otherwise service will be switched to the traffic flow with a minimum priority index.

4 Model implementation

To investigate the effectiveness of the above-mentioned traffic control algorithms, a traffic simulation model was built specifically for the study which allows for the implementation of the various algorithms and provides data pertaining to the algorithmic performances and the abilities of the algorithms to reduce vehicle waiting times and queue lengths. The model was designed and implemented using the software suite AnyLogic 6.8 [7] and utilises agent-based modelling techniques, or, more specifically, the modelling and simulation of systems that consist of autonomous, interacting individual agents.

5 Results

The results presented in this section are a proof of concept illustrating the potential of self-organising traffic control algorithms. They were obtained for a single, four-way intersection, with two lanes approaching from each direction. The inter-arrival times of vehicles to the system were randomly generated according to an exponential distribution with parameter λ, which is equivalent to generating the arrival times according to a Poisson distribution with an arrival rate of λ vehicles per second. [3]. The maximum speed that each vehicle attempts to obtain depends on the vehicle type and is determined according to a function of the predetermined speed limit for the road section. The vehicle following distances are determined according to a function of the vehicle's speed. All vehicle accelerations and decelerations are assumed to be constant.

The results presented compare the average queue lengths as well as the average time spent by vehicles in the system for three different control strategies for five different values of λ, ranging between 0.1 and 0.5. Each simulation run was the equivalent of 12 hours and was preceded by a 10 minute warm-up period. The first control strategy tested was an optimised fixed-time cycle-based strategy, in which the optimal green time[1] is implemented for each value of λ in terms of minimising the average queue lengths or the average time spent in the system by vehicles. The second control strategy implemented was the self-organising traffic control algorithm introduced by Lämmer and Helbing [2] (referred to here as SOTCA I[2]), and uses π_i^I in (1) as the priority index. The third control algorithm implemented was the traffic

[1] An optimal green time was found by running the simulation repeatedly, keeping the vehicle arrival rate fixed while varying the green time of the cycle. The green time which resulted in the best value for each respective performance measure being investigated was taken as the optimal green time for that specific value of λ.

[2] SOTCA is an acronym for Self-Organising Traffic Control Algorithm

control algorithm introduced in this paper (referred to as SOTCA II) and uses the priority index π_i^{II} in (3). The average queue lengths and time spent by vehicles in the system are shown in Figures 2 and 3, respectively.

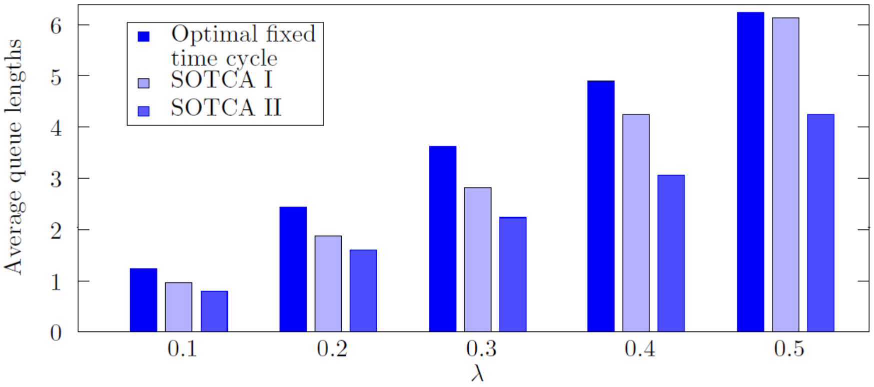


Figure 2: Average queue lengths.

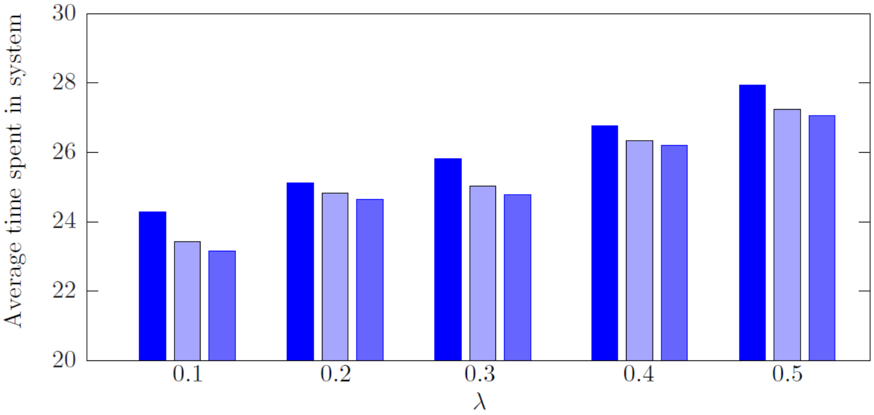


Figure 3: Average time spent by vehicles in the system.

implementing a maximum allowable green time, which when reached, will result in an automatic switching of service. Thus, the algorithm serves all incoming traffic as quickly as possible according to the switching strategies it determines, unless the amount of green time allocated to clear a waiting queue exceeds a certain allowable maximum, in which case said maximum allowable green time is implemented. From both graphs, it may be seen that both SOTCA I and SOTCA II outperform the optimised fixed-time cycle-based algorithm for each value of λ. This may be attributed to the flexibility of the self-organising traffic control algorithms as they are able to utilise any free intersection capacity to serve those vehicles requiring service most urgently, instead of waiting for a fixed amount of time to pass before switching service. It may also be seen the SOTCA II outperforms SOTCA I for each value of λ. This may be attributed to the fact that SOTCA II utilises real-time traffic data in order to best determine the most efficient and effective green times in order to optimise traffic flow, and while the differences in this instance are not considerable, they are expected to become more pronounced when the traffic control algorithms are implemented on a larger traffic network with more intersections, due to both improved traffic control at the intersections themselves as well as the improved formation and propagation of green-wave movements of platoons of vehicles through the system. The performance of SOTCA II is also expected to improve relative to the other two algorithms when more complex phase compositions are used (*e.g.* when exclusive turning phases are introduced) as the radar detection technology together with the algorithm logic will be able to determine whether or not certain phases can be "skipped" due to a lack of demand in terms of vehicle requirement.

6 Conclusions and future work

From the above results it is clear to see that there is potential in applying self-organisation techniques to traffic control, as is illustrated by the improvement shown by the self-organising control techniques over the optimised fixed time-cycle based technique, in terms of minimising queue lengths and reducing the amount of time spent in the system by vehicles. The aim for the future of this research project is thus to improve on the algorithms introduced in this paper, as well as conducting further research into novel traffic control algorithms which make use of the data provided by the radar detection equipment in real time. It is thought that clustering techniques may be employed to group the vehicles together into platoons so that the algorithms may focus on platoons of vehicles rather than on individual vehicles in an attempt to better facilitate the propagation of green waves of vehicles through adjacent intersections. It would also be desirable to experiment with these traffic control algorithms on larger, and more complex road network topologies.

Bibliography

[1] International Business Machines, 2007, *The roads to a smarter planet*, [Online], [Cited: 8 June 2012], Available: `http://www.ibm.com/smarterplanet`

[2] Lämmer S and Helbing D, 2008, *Self-control of traffic lights and vehicle flows in urban road networks*, Journal of Statistical Mechanics: Theory and Experiment, **2008**, p. P04019.

[3] Ross SM, 2006, *Simulation*, Fourth edition, Elsevier, San Diego (CA).

[4] SCOOT, 2009, *How SCOOT works*, [Online], [Cited: 8 June 2012], Available: `http://www.scoot-utc.com/documents/1_SCOOT-UTC.pdf`

[5] Serugundo GDM, 2004, *Self-organisation: Paradigms and applications*, Lecture Notes in Computer Science, **2997**, pp.1–19.

[6] Traffic Management Technologies, 2006, *Wavetronix SmartSensor Advance Model 200*, [Online], [Cited: 8 June 2012], Available: `http://www.tmtservices.co.za/changes2/images/products/TMT_WT_SSADV.pdf`

[7] XJ Technologies, 2010, *AnyLogic help*, [Online], [Cited: 8 June 2012], Available: `http://www.xjtek.com/`

Proceedings of the 2012 ORSSA Annual Conference
pp. 107–116

www.orssa.org.za/wiki/uploads/Conf/2012ORSSAConferenceProceedings.pdf

ORSSA Proceedings
ISBN
978-07972-1406-4
© 2012

Studies in Metaheuristics for the Blood Assignment Problem

E Dufourq[1] MO Olusanya[2] AO Adewumi[3]

Abstract

There is a daily and continuous demand for blood transfusion in hospitals and other medical outfits. However, units of blood do not have an unlimited shelf life and hence must be assigned to patients rapidly. Moreover, the natural blood grouping puts constraints on the type of blood that can be transfused to individuals. In case where blood banks cannot meet the continuously demands, they result to importing blood products from other source. This however might have serious implication in case of emergency hence there is need for proper management of available blood in stock in order to minimize import from outside. This paper presents a comparative study of different metaheuristics for the blood assignment problem. We present the result from genetic algorithms (GA), hill climbing and simulated annealing with several mutation operators designed and tested for the GA. Results obtained show that GA outperforms other the allocation process.

Key words: Blood component, blood assignment, optimization, genetic algorithms, hill climbing, simulated annealing

1 Introduction

Each day, there are thousands patients who require blood transfusion for surgical operations or other medical reasons. Blood transfusion which consists of transferring blood into the circulatory system of a person is therefore a daily practice in almost all countries of the world. Blood however is a limited resource which supply is limited by the amount donated and stored in the blood bank [1]. Moreover, the management of blood is further constrained by its limited shelf life and availability [1]. Donations are performed voluntarily and it is a common procedure to remove approximately one pint of blood from the donor. Once a donor has given blood they cannot give blood again for a certain period of time, which is one of the reasons for its scarce.

Generally, blood consists of several components including the white blood cells and the red blood cells [2]. This study focuses only on the red blood cell component of the blood. According to the ABO blood group system [2][3], the four main blood types are the A, B, AB and O. The presence of certain antigens and antibodies in each blood type determines the compatibility of each blood type with another hence transfusion must be done according to established compatibility. It is therefore vital that patients are assigned the correct blood type when requested, as mixing incompatible blood groups can leads to blood clumping or agglutination, which is dangerous for these patients. The blood type compatibilities are complicated further by the Rhesus factor (Rh) [2] and this doubles the number of blood grouping into A+, A-, B+, B-, AB+, AB-, O+ and O- (see [2][3][4] & [5] for more information on blood compatibility).

[1] University of KwaZulu-Natal, South Africa, email: edufourq@gmail.com
[2] University of KwaZulu-Natal, South Africa, email: soji˙sanya@yahoo.com
[3] Corresponding author: University of KwaZulu-Natal, South Africa, email: adewumia@ukzn.ac.za

In a blood bank therefore, there are daily requests for blood as well as number of donations, which is the main source of blood. The blood assignment problem (BAP) therefore consists finding an optimal way of allocating available blood of various types from donors to requesting patients in order to minimize the number of blood import from outside the blood bank while ensuring that compatibility and other constraints are satisfied. As stated earlier, when the blood bank does not have sufficient amount of blood for some type available, it has to import the makeup quantities from external source. Importing blood from outside the system is usually more expensive and might not meet emergency cases. The assignment problem also takes into consideration the lifetime of the units of blood. Hence the main objective of the BAP model is to meet all the requests for a particular day and to minimize the number of units imported from outside as well as to minimize the number of expired units.

This study is organized as follows: Section II discusses the theoretical model used to solve the blood assignment problem, Section III presents the methodology used to solve the problem, Section IV illustrates the experimental setup and design for our simulation, Section V presents results obtained and discusses them, Section VI highlights further areas of improvements and study, and finally Section VII contains the conclusion.

2 Theoretical Model

Various forms of the BAP with different objectives, models and solution techniques have been considered in literature. Techniques employed include the use of linear programming [6], stochastic programming [7], and simulations [8], among others. In most cases the objective is mainly to minimize imports and wastage.

The model used in this study is closely related to the model provided in Angelis *et al.* [4]. The model operates over a number of days. At each day t, there are a certain number requests for each unit of blood, that is, A+, A-, O+, and so on. The assumption is that if this demand cannot be met for day t, then blood will have to be imported from outside the system for that day. The blood is then assigned for the specific day. Donations for the day are added at the end of each day. Any left over for day t, that is the remaining volume of blood are carried over into the volume for day t+1. However if on day t certain units of blood expire then they are to be removed from the blood bank immediately. The model presented in [4] categorizes the requests into very urgent, urgent, and not urgent. This aspect is not considered in this study, hence once a request is made on day t, the deadline for that request to be satisfied is day t itself. This represents a hard constraint for our model, that is, all daily requests must be met.

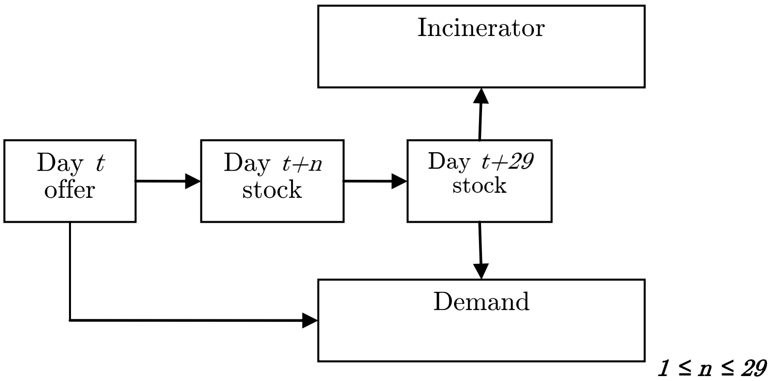


Figure 1: *Life cycle. Adapted from [4]*

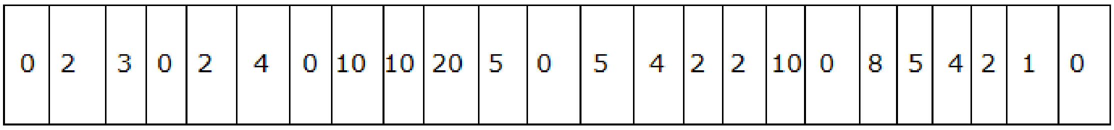
The life cycle of a unit of blood in terms of days is illustrated in figure 1. The model is similar to that of [4] with a slight difference. For a more realistic model, an external factor is considered for each day t. This takes care of any possible emergency that can drastically increase the demand beyond the daily baseline demand. In such a case, the blood bank may or may not have sufficient supply to meet this drastic change in demand. This concept was incorporated into the system.

Angelis *et al.* [4] developed a multi-objective, multi-period, multi-product linear programming model to represent the flow of blood in and out of the system. However their model dealt with only one blood types at a time. The model developed in this study considers all the blood types at once for a given request and uses blood compatibility to meet requests based on the *value* of the blood. Base on the known natural blood compatibility (see [1][2][5]), blood type O- is a universal donor that can be transfused into any patient hence considered more valuable other like type AB- which can only be transfused into a patient of the same type. Hence, when there is a demand of a specific type, say AB-, it could be considered wise to allocate more units of AB- than to allocate units of O- with the hope of preserving "universal" blood types. Based on the compatibility constraint, we have developed a vector representation with *value* allocated to indicate the proportion of type transfusion that can be made to other types. The compatibility constraint naturally enforces us to have 27 possible transfusion possibilities. Hence, if pick A+ as a type, we can only transfuse A+ to either A+ or AB+ hence the A+ has a count of 2 and a value of 2 out of 27. Table 1 presents the values for each type of blood used in the system. The goal of this paper is to study the performance of heuristics for the BAP since literature has earlier reported the use of exact method for a limited version of the problem [4]. We adopt a vector representation for our heuristics based on the design explained in Table 1. Vector 1 shows how an allocation in the model is represented while vector 2 illustrates an example of how this vector is interpreted.

Table 1: *Blood Type Value*

Type	Allocate to	Count	Value
A+	A+, AB+	2	2/27
A-	A+, A-, AB+, AB-	4	4/27
B+	B+, AB+	2	2/27
B-	B+, B-, AB+, AB-	4	4/27
AB+	AB+	1	1/27
AB-	AB+, AB-	2	2/27
O+	A+, B+, AB+, O+	4	4/27
O-	All	8	8/27

Vector 1: *Sample representation of an allocation*



0	2	3	0	2	4	0	10	10	20	5	0	5	4	2	2	10	0	8	5	4	2	1	0

Vector 2: *Interpretation of chromosome representation*

From blood type	A+	A+	A-	A-	A-	A-	B+	B+	B-	B-	B-	B-	AB+	AB-	AB-
To blood type	A+	AB+	A-	A+	AB-	AB+	B+	AB+	B+	B-	AB+	AB-	AB+	AB+	AB-

Vector 2 *continued: Interpretation of chromosome representation*

From blood type	O+	O+	O+	O+	O-	O-	O-	O-	O-	O-	O-	O-
To blood type	O+	A+	B+	AB+	A+	A-	B+	B-	AB+	AB-	O+	O+

Hence from vector 1, the first element signifies that no blood was allocated from A+ to a request of A+. The second element signifies that 2 units of A+ were allocated to the request of AB+ and so on. From the above it is clear that the model aims at optimizing the allocations from a particular blood type to corresponding types, hence we can consider the problem as a form of knapsack problem.

The knapsack problem [9,10] is an NP hard combinatorial optimization problem that involves placing a number of items into a knapsack or container. Each item has a certain value and weight. The multiple knapsack problem (MKP) has more than one container, each with varying capacity [10]. The BAP can then be conceived as a MKP where units of blood are to be allocated to requests of different types (sizes). However, in our model, the goal is to minimization, where a better allocation is one who has smaller *value*. For example a unit of O- has a value of 8/27 since it is a "universal" donating type and hence the system should attempt to use as little units of O- as possible. The aim is to maintain as much volume of higher *value* blood in the blood bank to cater for emergency hence minimizing importation from outside the bank. Thus the objective of our model is to minimize imported blood while also minimizing the total *value* of an allocation. This must be done such that all demands are met whilst attempting to avoid wastage from expired units. Thus the objective function can be put as

$$\text{Minimize} \qquad \sum_{t=1}^{n} I(t)$$

where $I(t) = I_{o-}(t) + I_{o+}(t) + I_{A-}(t)\, I_{A+}(t)\, {}_{+} I_{B-}(t) + I_{B+}(t) + I_{AB-}(t) + I_{AB+}(t)$

I(t) is the total number of blood imported from outside in day t which consist of all units of import blood of types, O-, O+, A-, and so on.

At all times the system keeps track of numerous things. Firstly the daily request, the time remaining for units on the shelf, the number of donations at the end of the day and the current amount of units in the bank. Certain heuristics are used to determine the best allocation method for each request given the amount of units in the bank for that particular day. These heuristics are discussed in the section below.

3 Methodology

3.1 Genetic Algorithms

Genetic algorithm (GA) has been widely applied to solve combinatorial problems. It simulates certain features of natural evolution, by evolving chromosomes defined as "organic devices for encoding the structure of living beings" [11]. GAs uses recombination and mutation to create new chromosomes similar to biological reproduction. Recombination takes features from the parents and adds those features to their children. Mutation on the other hand alters genes in the chromosome. GA aims to evolve the best possible chromosome to solve a problem at hand. In this study, the chromosome was represented as in vector 2. Several variations of operators were used to determine which operator is most suitable for the chromosome created. Literature presents a number of operators designed for GAs [11]. However, as in other permutation based problems like traveling salesman, generic evolutionary operators might produce undesirable and infeasible results for the BAP hence we try to designed our operator for the GA. A very brief discussions on operators used in this study are given below.

1) *Single point crossover*

Single point crossover randomly selects a point in the chromosome then splits the parents at the point. All remaining genes after that point are interchanged between the two parents. This is the most basic form of crossover, which we also adopted for our problem.

2) *Blood type crossover*

In blood type crossover, a type of blood is randomly select from the 8 types available. Then the genes of that type in the "from" section of the chromosome (see Vector 2) are interchanged between the two parents.

Consider vectors 3 and 4 and suppose blood type A+ was randomly selected. Genes affected by A+ are represented by the greyed out boxes. Those genes are then interchanged to produce the children shown in vectors 5 and 6. All remaining genes remain unchanged.

Vector 3: *Parent 1*	0	2	3	0	2	4	0	10	10	20	5	0	5	4	2	2	10	0	8	5	4	2	1	0
Vector 4: *Parent 2*	1	2	2	3	4	2	1	0	2	4	5	4	5	5	4	4	2	2	2	2	2	1	1	5
Vector 5: *Child 1*	1	2	3	0	2	4	0	10	10	20	5	0	5	4	2	2	10	0	8	5	4	2	1	0
Vector 6: *Child 2*	0	2	2	3	4	2	1	0	2	4	5	4	5	5	4	4	2	2	2	2	2	1	1	5

3) *Mutation on volume*

The first type of mutation used is the most basic mutation found in literature [11] which alters each gene independently based on some probability of mutation. If a gene is to be mutated, its new value is randomly generated between 0 and the volume of blood available for that specific type. For example consider the first gene that represents blood type A+, if there are 10 units of blood available for A+, then a random number between 0 and 10 is generated and inserted into the child at that position. The same process applies to each gene.

4) *Mutation on request*

This form of mutation is similar to the one described above with the difference that the new value is randomly generated based on the number of requests for the specific blood type. For example, if the second gene which allocates blood from A+ to AB+ is selected, a random number from 0 to the number of requests of AB+ is generated in place of the method in mutation by volume.

5) *Swap mutation*

This type of mutation is similar to mutation for chromosomes having permutation representations. For example, two random position of genes are selected and their value are swap as shown in the illustration below.

0	2	3	0	2	4	1	10

0	1	3	0	2	4	2	10

6) *Blood type mutation*

Blood type mutation swaps genes based on their blood type. Each gene is selected and swapped with another gene of same type. So when the first gene is examined, it can be swapped with genes at position 0, 3, 16 or 19 (where the first gene has position 0).

3.2 Hill Climbing

Hill climbing (HC) is a local search optimization technique [12]. The algorithm makes stepwise moves that evaluates each solution in turn and only accepts those which are better than the current solution. It is a single solution based technique hence in our study a single chromosome as designed above was used with no crossover, selection or mutation operators as in GA. HC algorithm works on a current solution to find the next better solution. More information on the HC can be found in [12].

3.3 Simulated Annealing

Simulated annealing (SA) [12] works on the analogy of annealing in metallurgy. The concept is to heat a material to some high temperature and let it cool down gradually to form a crystal. The process of cooling down affects certain properties in the material. When the temperature is high particles within the material move rapidly around and once lower temperatures are reached the particles slow down until they reach a final state. Again, the same vector representation has been used in SA as in GA and HC. At high temperatures, chromosomes with high valued functions (weaker chromosomes) are accepted more frequently while at lower temperatures only chromosomes have improved solutions are accepted.

The probability of acceptance of poorer solution is based on equation (1). At low temperatures, the SA in this study performs a certain number of extra iterations in order to further exploit the solution space.

$$random[0,1) < e^{\frac{\mathrm{currentSolution} - \mathrm{newSolution}}{\mathrm{temperature}}} \qquad (1)$$

The chances of accepting poorer solutions make GA better than the HC [12]. HC only accepts better solutions and hence may get stuck in a local optimum and remain there for the duration of the run. SA allows worst solutions to be accepted and hence can get out of a local optimum.

4 Results and Discussion

4.1 Experiment Setup

Simulation experiment was performed on a regular PC with average configuration running in a Microsoft Window environment. Programs were written in Java and run in an Eclipse SDK IDE 3.7.1 environment. Since this is a test study for metaheuristics, we assume a period of 10 days of requests. This can easily be extended for real-life case of longer period. Due to difficulty in obtaining real-life data, requests and donations for blood were randomly generated based on the blood type distribution in a representative population. Moreover, there is no study aside [5], to the best of our knowledge, that have evaluated the use of GA, HC and SA for our model of BAP hence we decided to set our own benchmark parameters.

The environment consisted of 500 individuals. The population size for GA was set at 1000 and the number of generations at 125. The number of iterations for HC was set at 900. The initial temperature for SA was set at 45 with an increment temperature at of 0.0005 and the final temperature of 0.05. In our SA simulation, an additional mutation is performed once the algorithm reached a temperature below 2.5.

Each algorithm was run using the above parameter setting. Each algorithm was run for a total of 15 times and the average results obtained were recorded. In order to determine the best algorithm and the best of the proposed operators for the BAP, each operator was tested on each algorithm. Hence GA was tested using both crossovers and the 4 mutation operators, HC and SA were tested using all 4 mutation operators. Except where otherwise stated, result of GA is that obtained by combining 1-point crossover with the mutation by volume operator. Since internal workings of the algorithms are different and the fact that results obtained rather than time taken is the critical factor to this study, we excluded the measure of simulation time for all algorithms. For example, HC is generally known to generate results, whether good or bad, than GA which has a lot of parallel computations.

4.2 Results and Discussion

4.2.1 Genetic Algorithms

Tables 2, 3 and 4 present the results obtained using GA. Table 2 shows that the blood type crossover performs nearly as well as the common 1 point crossover, both in combination with mutation by volume. In both cases the most blood was imported from blood type O+. Both operators resulted in no units of blood of type AB+ being imported which is a positive feedback since AB+ can receive blood from any group and hence it would not make sense if there were many imports for that type.

Table 3 shows comparison of the mutation by volume and mutation by request operators, in combination with the 1 point crossover operator. The performance almost ranked similar. Table 5 however clearly demonstrates that mutation generation via random gene interchange results in a much higher cost. This is probably due to the random nature of this operator than simply swapping genes around without any decision taking place.

Table 2: *GA 1 point vs Bloot Type crossover*

Imports	1-point CrossO	Blood type CrossO
A+	44.4	32.13333
A-	9.333333	12.33333
B+	16.4	10.33333
B-	2.733333	3.8
AB+	0	0
AB-	0.2	0.066667
O+	144.1333	182.8667
O-	45.73333	48.8
Sum	262.9333	290.3333

Table 3: *GA mutation based on volume vs mutation based on request*

Imports	Mutation (volume)	Mutation (request)
A+	44.4	58.87
A-	9.33	13.67
B+	16.40	24.73
B-	2.73	1.87
AB+	0.00	0.00
AB-	0.20	0.47
O+	144.13	106.47
O-	45.73	39.80
Sum	262.93	245.87

Mutation by swapping blood types did not perform too well. Although, there was an improved performance when compared to the mutation via random interchange, it is not of much significance. Clearly therefore, the newly designed problem-based mutation volume or request are recommended for use than generic operator with GA.

Table 4: *GA mutation based on random vs mutation based on Type*

Imports	Mutation (random)	Mutation (Type)
A+	579.64	354.67
A-	47.73	82.67
B+	113.64	124.92
B-	14.18	33.50
AB+	5.82	17.92
AB-	2.55	12.92
O+	727.45	489.00
O-	104.45	177.17
Sum	1595.46	1292.75

All the four mutation operators however did have the imports of blood type AB+ relatively low. Using volume and request both ended importing no units at all for AB+ as opposed to mutation via random interchange and by blood type. The importation of O- is not the highest (compared to other blood type) in any of the operators which is a positive feedback considering the fact that O- is the most valuable type (universal donor).

4.2.2 Hill climbing

Tables 5 and 6 present the results obtained for HC. At a first glance the results follow a similar pattern to that obtained by GA in the sense that the number of imports for blood type AB+ remains relatively low in comparison to other imports.

Once again it can be observed that mutation via random generation produced the worst results. However, the values obtained are significantly better than the mutation with random generation which the GA displayed.

Table 5: *HC mutation based on volume vs mutation based on request*

Imports	Mutation (volume)	Mutation (request)
A+	15.00	42.53
A-	23.47	30.33
B+	11.07	14.00
B-	6.20	14.00
AB+	0.33	2.533
AB-	0.20	3.13
O+	201.00	180.87
O-	76.33	76.13
Sum	333.60	363.53

Table 6: *HC mutation based on random vs mutation based on Type*

Imports	Mutation (random)	Mutation (Type)
A+	114.80	35.27
A-	27.80	32.07
B+	40.07	27.53
B-	13.53	12.40
AB+	1.60	8.27
AB-	3.267	3.27
O+	382.53	304.40
O-	170.40	184.80
Sum	754.00	608.00

4.2.3 Simulated annealing

SA performed nearly as well as the other two however the results were slightly higher. As expected, SA took longer to run than other method but the longer computational time did not yield improved results.

Table 7: *SA mutation based on volume vs mutation based on request*

Imports	Mutation (volume)	Mutation (request)
A+	77.73	32.55
A-	14.20	11.18
B+	19.33	7.55
B-	4.20	4.27
AB+	0.07	0.00
AB-	0.33	0.00
O+	234.73	106.36
O-	45.80	47.64
Sum	396.40	209.55

Table 8: SA mutation based on random vs mutation based on Type

Imports	Mutation (random)	Mutation (Type)
A+	169.20	63.80
A-	28.07	27.07
B+	47.07	45.00
B-	11.67	13.87
AB+	2.13	16.73
AB-	2.20	4.13
O+	436.93	370.73
O-	156.87	155.93
Sum	854.13	697.27

Table 9 shows the results of GA and HC extracted from results obtained in [5]. In both GA and HC, a huge amount (60%) of blood of type O- was imported into the system. O- , the universal donor, thus has much greater *value* than say A+ since it is compatible with more blood types than A+. The current study has however, improved on this previous results of [5] with only 17.3% of blood type O- being imported from outside. Furthermore, the GA reported in [5] did not import much blood of specific types, like A+, and B+.

Table 9: *GA and HC [5]*

	Result from [5]			
	GA	**%**	**HC**	**%**
A+	0	*0*	0	*0*
A-	37	*12.7*	38	*13.1*
B+	0	*0*	0	*0*
B-	10	*3.4*	10	*3.4*
AB+	0	*0*	0	*0*
AB-	0	*0*	0	*0*
O+	69	*23.7*	61	*20.9*
O-	175	*60*	182	*62.5*

Table 10: *GA and HC from this study*

	Current study			
	GA	**%**	**HC**	**%**
A+	44	16.9	15.00	*4.4*
A-	9	3.4	23.47	*7.0*
B+	16	6.1	11.07	*3.3*
B-	2	0.7	6.20	*1.8*
AB+	0	0	0.33	*0*
AB-	0	0	0.20	*0*
O+	144	55.3	201.00	*60.2*
O-	45	17.3	76.33	*22.8*

Similarly, the HC reported in [5] did not have an even distribution of the blood imports, rather up to 62.5% of imports came from O- as opposed to 22.8% from this study.

5 Conclusion

In this paper, we have presented a successful and improved application of GA, HC, and SA to solve the BAP. The BAP model is a modified version of that obtained in literature which aims at minimizing blood importation from outside the blood bank while ensuring that daily demands are also met. We presented some problem-specific operators for crossover and mutations were designed for our algorithms which proved to give better results than that obtained in an earlier study. Results of GA outperform that of SA and HC. The GA however did provide a better distribution of blood importation with fewer imports from more valuable types. We hope to further improve on the designed operator and study their impact on the algorithms. Efforts will also be taken to test the limit of each algorithm as the problem space increases in comparison with exact solution approach. The design, once perfected, will be applied to real-life situation and data.

Bibliography

[1] Charpin, J.P.F and Adewumi, A.O. (2011). Optimal Assignment of Blood in Blood Banking. *Proceedings of the Mathematics in Industry Study Group (MISG 2011).* January 9-14.

[2] South African National Blood Service, What's your type. Available: http://www.sanbs.org.za.[Accessed on 3 October 2012].

[3] American Red Cross (2012). Blood Type. Available: http://chapters.redcross.org/br/northernohio/INFO/bloodtype.html [Accessed on 27 May 2012].

[4] Angelis, D. V., Ricciadrdi, N., and Storchi, G. (2001). "Optimizing Blood Assignment in a Donation Transfusion System". *International Transactions in Operational Research.* 8(2):183-192

[5] Adewumi A.O., Budlender N., and Olusanya M.O. (2012). Optimizing the Assignment of Blood in a Blood Banking System: Some Initial Results. *CEC 2012, IEEE World Congress on Computational Intelligence*. Brisbane, Australia, June 10-15.

[6] Kendall, K. E., and Lee, S. M. (1980). Formulating Blood Rotation Policies with Multiple Objectives. *Management Science.* 26: 1145-1157

[7] Sapountzis, C. (1989). Allocating Blood to Hospitals. *The Journal of the Operational Research Society.* 40: 443-449.

[8] Katsaliaki, K., and Brailsford, S. C. (2007). Using Simulation to Improve the Blood Supply Chain. *The Journal of the Operational Research Society.* 58(2) : 219-227

[9] Martello, S. and Toth, P. (1990). *Knapsack Problems, Algorithms and Computer Implementations*. University of Bologna, Italy.

[10] Pisinger, D. (1995). *Algorithms for Knapsack Problems, PhD thesis*. Department of Computer Science. University of Copenhagen, Denmark, February 1995.

[11] Davis, L. (1991). *Handbook of genetic algorithm*. Van Nostrand Reinhold.

[12] Weise, T. (2009). *Global Optimization Algorithms - Theory and Application*. Electronic book. June 2009.

Proceedings of the 2012 ORSSA Annual Conference
pp. 117–124

www.orssa.org.za/wiki/uploads/Conf/2012ORSSAConferenceProceedings.pdf

ORSSA Proceedings
ISBN
978-07972-1406-4
© 2012

A Survey of Hyper-Heuristics for the Nurse Rostering Problem

N Pillay[1] C Rae[2]

Abstract

The first international nurse rostering competition held in 2010 has stimulated new interest in this domain. Various techniques such as tabu search, genetic algorithms, simulated annealing, integer programming and constraint programming have been applied to solving the nurse rostering problem. Hyper-heuristics, a fairly new approach, searches a heuristic space rather than a solution space in an attempt to provide a more generalised solution to a problem. There are essentially four categories of hyper-heuristics, namely, selection constructive, selection perturbative, generation constructive and generation perturbative. This paper provides an overview of the use of hyper-heuristics for solving the nurse rostering problem, and based on a critical analysis of the application of hyper-heuristics to this domain proposes directions for future research.

Key words: hyper-heuristics, nurse rostering

1 Introduction

The nurse rostering problem involves assigning nurses to shifts in such a manner that the constraints of the problem are met. There has been a fair amount of research into solving this problem [5]. Hyper-heuristics is a novel approach that has been successful at providing generalised solutions to various combinatorial optimization problems such as university examination timetabling, production scheduling and packing problems [7]. There has not been much work investigating the use of hyper-heuristics for solving the nurse rostering problem. This paper aims to provide a foundation for further development of this field by firstly providing an account of the research done to date in this area and suggesting directions for growth based on a critical analysis of previous work.

The following section describes the nurse rostering problem. Section 3 provides an overview of hyper-heuristics. An account of research into the use of hyper-heuristics to solve the nurse rostering problem is presented in section 4. Section 5 proposes directions for future research based on a critical analysis of previous work.

2 The Nurse Rostering Problem (NRP)

The nurse rostering problem falls into the NP-complete category of personnel scheduling problems which, due to its potential real-world applications, has been studied over the past 40 years. De Causmaecker et al. [13] describe this as the most difficult of the personnel scheduling

[1] Corresponding author: University of KwaZulu-Natal, School of Mathematics, Statistics and Computer Science, South Africa, Private Bag X01, Scottsville, 3209, Pietermartizburg, email: edufourq@gmail.com
[2] University of KwaZulu-Natal, School of Mathematics, Statistics and Computer Science, South Africa, email: sanya@yahoo.com

problems. The NRP essentially involves assigning a given set of shifts, which are defined as working periods, to the available nursing staff, subject to a set of constraints [1]. The nursing staff possess different skills and shifts must be assigned to nurses that have the skills needed for the particular shift [17]. The constraints are generally set by the hospital and the preferences of the nursing staff [14]. A typical problem specification for an NRP usually includes the number of nurses; skills of the nurses; availability of nurses; total number of assignments per nurse; the shift types, e.g. day, night; minimum and maximum number of working hours per week; minimum or maximum number of consecutive assignments; minimum or maximum number of free intermediate shifts/days; the number of nurses required per shift; the minimum and maximum number of working weekends; nurse shift preferences; ward requirements; overtime requirements and other constraints such as shift compatibility, e.g. no night shift before a weekend; workload equity and unwanted shift patterns [11, 13, 15, 17]. The constraints are divided into two categories, namely, hard constraints and soft constraints. Hard constraints are those that must be met by a roster. A roster satisfying all of these constraints is referred to as feasible. Examples of hard constraints include the required number of nurses possessing the necessary skill must be allocated for each shift and a nurse can only work one shift per day. The following equation represents the sum of the total number of violations m for n hard constraints:

$$HC = \sum_{i=1}^{n} \sum_{j=1}^{m} h_{ij}$$

A value of zero indicates a feasible timetable.

Soft constraints are satisfied as far as possibly, but a roster produced is still considered operable even if all the soft constraints are not satisfied [11]. Thus, the aim is to minimise the number of soft constraints violated. The soft constraint cost is used as a measure of the quality of the timetable. Typical soft constraints include nurse preferences; minimum and maximum assignments per nurse; minimum and maximum consecutive free days; and unwanted shift patterns amongst others. Those constraints that are defined as hard or soft differs from one NRP to the next. The following equation representing the number of soft constraints violations p for q soft constraints is minimized:

$$SC = \sum_{i=1}^{q} \sum_{j=1}^{p} s_{ij}$$

The aim is to produce a periodic duty roster that does not violate hard constraints and minimizes soft constraint violations. Creating such schedules has usually been done over the course of many days by hand [17]. While this tends to work, it is a time consuming activity that allows for very little flexibility and staff are not always satisfied with the roster produced [5]. For this reason a wide range of approaches have been used to tackle this problem domain, such as integer programming [5], linear programming [5], constraint programming [5], genetic algorithms [1, 5, 17], tabu-search [5, 15], simulated annealing [5], heuristic techniques [5] and expert systems[5]. Mathematical programming techniques have been found to be unable to cope with the large search spaces of modern problems, linear and integer programming in particular are unable to satisfy a wide range of constraints sufficiently [5].

In order to compare the performance of the various techniques in solving the nurse rostering problem, benchmark data sets have been created. De Causmaecker et al. [13] provide a summary of the different benchmark sets. The data comprising these sets are either computer generated or have been gathered from real-world hospitals. The most recent benchmark set is that used for the first international nurse rostering competition [15]. The data sets have been modeled on real-world nurse rostering problems. The problems are divided into three categories, namely, sprint, medium and long analogous to Olympic distances. The sprint, medium and long problems must be solved in a few seconds, minutes and hours respectively.

Most of the research in this domain has resulted in methods producing the best results for one or more data sets being identified. However, these techniques lack generality in finding solutions over multiple instances of the nurse rostering problem. This is the focus of hyper-heuristics.

3 Hyper-Heuristics

Hyper-heuristics aim at providing a more generalized solution to a problem by searching a heuristic space instead of a solution space [6]. This is achieved by either selecting low-level heuristics or combinations of low-level heuristics. Alternatively, a hyper-heuristic can generate low-level heuristics and genetic programming has been used for this purpose [8]. Genetic programming is a variation of genetic algorithms which searches a program space instead of a solution space. The low-level heuristics can be constructive or perturbative. Construction heuristics are used to create a solution to a problem. For example, in the domain of examination timetabling construction low-level heuristics generally used are the largest degree, largest weighted degree, saturation degree, largest enrollment and largest colour degree heuristics [7]. These heuristics are a measure of the difficulty of scheduling an exam and are used to decide which examination should be scheduled next. Perturbative low-level heuristics are used to improve an initial solution to a problem. For example for the nurse rostering problem an initial roster can be created by randomly assigning shifts to nurses. Examples of low-level heuristics that can be used to improve the initial roster are swap shifts, de-allocate shifts and re-allocate shifts, amongst others.

Thus, hyper-heuristics can be divided into four main categories, namely, selection constructive, selection perturbative, generation constructive and generation perturbative [9]. Most of the research on hyper-heuristics have focussed on selection constructive and selection perturbative hyper-heuristics. Techniques employed by selection constructive hyper-heuristics include case-based reasoning, tabu search, variable neighbourhood search, genetic algorithms, simulated annealing and great deluge [8, 10]. Selection perturbative hyper-heuristics are comprised of two components, namely, one for heuristic selection and a second for move acceptance. Methods that have been employed for heuristic selection include simple random, random permutation, greedy, choice functions and reinforcement learning. Move acceptance has been achieved using techniques such as accept all moves, only accept improving moves, tabu search, simulated annealing, great deluge, Monte Carlo and late acceptance.

The application of hyper-heuristics to solving educational timetabling and packing problems has been widely researched [8, 9]. Other domains that hyper-heuristics have been applied to on a smaller scale include the Boolean satisfiability problem, the travelling salesman problem and production scheduling. There has not been much research into the use of hyper-heuristics to solve the nurse rostering problem.

4 Hyper-Heuristics for the Nurse Rostering Problem

This section reports on studies using hyper-heuristics to solve the nurse rostering problem. The only hyper-heuristics that have been implemented for this domain are selection perturbative hyper-heuristics. A couple of studies have also combined a selection perturbative hyper-heuristic with another search method to solve the NRP.

One of the earlier studies in this field is that conducted by Cowling et al. [13] which evaluates the use of selection perturbative hyper-heuristic in solving 52 NRP instances for a UK hospital. This was a very simple hyper-heuristic which used a choice function for heuristic selection and accepted all moves. The hyper-heuristic improved an initial solution created by randomly allocating shifts to nurses. Input to the choice function was the individual performance of the low-level heuristics. Nine low-level heuristics which generally change a nurse's shift pattern based on a specified condition, e.g. the change will decrease the hard constraint violations, the change will decrease the soft constraint cost, the change will decrease the number of soft constraints Hyper-heuristics aim at producing good solutions for problems generally for a particular domain instead of finding the best solution for one or more problems. This research contributes to the larger initiative of evaluating the effectiveness of hyper-heuristics in solving different combinatorial optimization problems and also contributes to the domain of nurse rostering by investigating a novel approach to solving this problem. The previous section has presented a survey of the studies conducted to evaluate hyper-heuristics in solving the nurse rostering problem. Based on this survey and a comparison of the progress made in researching the effectiveness of hyper-heuristics in other combinatorial optimization domains, this section presents directions for expanding this area further.

The previous section provided an overview of the research conducted into the use of hyper-heuristics to solve nurse rostering problems. All of this work has focussed on selection perturbative hyper-heuristics. In a majority of these studies low-level heuristics are randomly selected. A choice function has also been used for this purpose. The most popular method used for move acceptance is simulated annealing. This method was also found to perform better than the great deluge for move acceptance. In the domain of educational timetabling a variety of methods have been tested for both heuristic selection and move acceptance such as reinforcement learning, genetic algorithms, variable neighbourhood search and Monte Carlo methods amongst others. It would be interesting to perform a similar evaluation of these various techniques for the NRP domain as well. In a number of studies developing selection constructive hyper-heuristics for educational timetabling, a metaheuristic such as tabu search or genetic algorithms is used to explore a heuristic space of combinations of low-level construction heuristics which are used to construct a timetable. The study conducted by Burke et al. [10] take a similar approach with a tabu search employed to search the heuristic space. Hyper-heuristics employing metaheuristics, such as variable neighbourhood search and genetic algorithms, to explore a space of combinations of low-level perturbative heuristics has not been evaluated for solving the NRP and this is a potential area for future research. In this case each combination will contain low-level pertubative heuristics instead of construction heuristics. Genetic programming can also be investigated as a means of evolving programs for heuristic selection and move acceptance.

The low-level perturbative heuristics defined for this domain include assigning shifts, deleting shifts and swapping shifts amongst nurses. In most cases the swapping is stimulus based instead of all swaps being accepted. In stimulus-based swapping some criterion e.g. a decrease in the soft constraint cost, has to be met in order for the swap to be accepted. The derivation of other heuristics is another area that has not been investigated. An investigation into the use of a generation perturbative hyper-heuristic, which employs genetic programming to derive new low-level heuristics by combining existing heuristics, is a potential area of future research for this domain.

There appears to be no construction heuristics defined for the domain and initial solutions are created by randomly assigning shifts to nurses. This could be attributed to the nature of the problem domain which is different from domains like educational timetabling for which construction heuristics have been defined. The derivation of construction heuristics for this domain using human intuition will allow for further development of this field. The use of a generation constructive hyper-heuristic could be examined for this purpose. This hyper-heuristic could derive construction heuristics by combining variables representing values for characteristics of the problem and arithmetic, logical and selection operators. These construction heuristics would be a function of the variables representing the characteristics of the problem. For example if n represents the number nurses and s the number of shifts a simple example of a function representing a construction heuristic would be n*s.

Two of the studies presented combining a hyper-heuristic with other methods used to search the solution space have produced promising results. Hybridising the search of the heuristic and solution space in solving the NRP could produce good results for this domain and warrants further investigation. This has proven to be very effective for the domain of educational timetabling.

The effect that the set of low-level heuristics used has on the performance of a hyper-heuristic needs to be studied further as well as the identification of methods for choosing an optimal set of low-level heuristics. Furthermore, the effectiveness of combining constructive and perturbative hyper-heuristics should be investigated.

In all of the studies presented, the hyper-heuristic has been tested on a particular problem or set of problems. The hyper-heuristics need to be tested more widely on different problem sets to ascertain their ability to generalise. There are four different benchmark sets that can be used for this purpose [13, 15].violated without increasing the hard constraint cost, were available for use.

Biligan et al.[4] employ a selection perturbative hyper-heuristic to solve the Belgian nurse rostering problem. The hyper-heuristic randomly selects perturbative heuristics and either simulated annealing or great deluge is used for move acceptance. The set of low-level heuristics was comprised of six low-level heuristics that assign shifts, delete shifts and change shifts. The hyper-heuristic employing simulated annealing for move acceptance produced the best results.

Burke et al. [11] used a tabu-search selection perturbative hyper-heuristic to create a nurse roster for a UK hospital. The hyper-heuristic improves an initial solution created by allocating shifts randomly to each nurse. This study uses the nine low-level perturbation heuristics introduced by Cowling et al. [13] described above. The use of a tabu list prevents heuristics that have not performed well from being applied for a set period. The hyper-heuristic was able to produce a feasible solution to the problem.

Hybrid approaches that combine a hyper-heuristic with another optimization technique have also been implemented. Biligan et al. [5] use a hybrid approach combining a hyper-heuristic and greedy shuffle heuristic to solve the benchmark problems for the first international timetabling competition. An initial feasible solution is randomly constructed. This initial solution is then improved using the hyper-heuristic. The hyper-heuristic randomly chooses low-level heuristics and simulated annealing is used for move acceptance. Twelve low-level perturbative heuristics are available. These heuristics essentially involve swapping the shifts of two randomly selected nurses. The roster produced by the hyper-heuristic is improved further by a greedy shuffle. The greedy shuffle swaps components of rosters for two nurses. Only swaps that produce feasible rosters of just as good or better quality are accepted. If there is no improvement after a set number of swaps the worst nurse roster is swapped with that of a randomly chosen nurse and the shuffle terminates.

Bai et al. [2, 3] use a selection perturbative hyper-heuristic and genetic algorithm hybrid to solve the NRP for a UK hospital. The genetic algorithm is hybridised with the hyper-heuristic and is applied to the solution space. It uses the crossover and mutation operators. The hyper-heuristic employs simulated annealing for move acceptance. The heuristic selection component chooses heuristics based on their performance up until the particular point of solving the problem. Acceptance ratios for heuristics are calculated for this purpose. The nine low-level heuristics used by Cowling et al. [11] described above are also used in this study.

5 Discussion

Hyper-heuristics aim at producing good solutions for problems generally for a particular domain instead of finding the best solution for one or more problems. This research contributes to the larger initiative of evaluating the effectiveness of hyper-heuristics in solving different combinatorial optimization problems and also contributes to the domain of nurse rostering by investigating a novel approach to solving this problem. The previous section has presented a survey of the studies conducted to evaluate hyper-heuristics in solving the nurse rostering problem. Based on this survey and a comparison of the progress made in researching the effectiveness of hyper-heuristics in other combinatorial optimization domains, this section presents directions for expanding this area further.

The previous section provided an overview of the research conducted into the use of hyper-heuristics to solve nurse rostering problems. All of this work has focussed on selection perturbative hyper-heuristics. In a majority of these studies low-level heuristics are randomly selected. A choice function has also been used for this purpose. The most popular method used for move acceptance is simulated annealing. This method was also found to perform better than the great deluge for move acceptance. In the domain of educational timetabling a variety of methods have been tested for both heuristic selection and move acceptance such as reinforcement learning, genetic algorithms, variable neighbourhood search and Monte Carlo methods amongst others. It would be interesting to perform a similar evaluation of these various techniques for the NRP domain as well. In a number of studies developing selection constructive hyper-heuristics for educational timetabling, a metaheuristic such as tabu search or genetic algorithms is used to explore a heuristic space of combinations of low-level construction heuristics which are used to construct a timetable. The study conducted by Burke et al. [10] take a similar approach with a tabu search employed to search the heuristic space. Hyper-heuristics employing metaheuristics, such as variable neighbourhood search and genetic algorithms, to explore a space of combinations of low-level perturbative heuristics has not been evaluated for solving the NRP and this is a potential area for future research. In this case each combination will contain low-level pertubative heuristics instead of construction heuristics. Genetic programming can also be investigated as a means of evolving programs for heuristic selection and move acceptance.

The low-level perturbative heuristics defined for this domain include assigning shifts, deleting shifts and swapping shifts amongst nurses. In most cases the swapping is stimulus based instead of all swaps being accepted. In stimulus-based swapping some criterion e.g. a decrease in the soft constraint cost, has to be met in order for the swap to be accepted. The derivation of other heuristics is another area that has not been investigated. An investigation into the use of a generation perturbative hyper-heuristic, which employs genetic programming to derive new low-level heuristics by combining existing heuristics, is a potential area of future research for this domain.

There appears to be no construction heuristics defined for the domain and initial solutions are created by randomly assigning shifts to nurses. This could be attributed to the nature of the problem domain which is different from domains like educational timetabling for which construction heuristics have been defined. The derivation of construction heuristics for this domain using human intuition will allow for further development of this field. The use of a generation constructive hyper-heuristic could be examined for this purpose. This hyper-heuristic could derive construction heuristics by combining variables representing values for characteristics of the problem and arithmetic, logical and selection operators. These construction heuristics would be a function of the variables representing the characteristics of the problem. For example if n represents the number nurses and s the number of shifts a simple example of a function representing a construction heuristic would be n*s.

Two of the studies presented combining a hyper-heuristic with other methods used to search the solution space have produced promising results. Hybridising the search of the heuristic and solution space in solving the NRP could produce good results for this domain and warrants further investigation. This has proven to be very effective for the domain of educational timetabling.

The effect that the set of low-level heuristics used has on the performance of a hyper-heuristic needs to be studied further as well as the identification of methods for choosing an optimal set of low-level heuristics. Furthermore, the effectiveness of combining constructive and perturbative hyper-heuristics should be investigated.

In all of the studies presented, the hyper-heuristic has been tested on a particular problem or set of problems. The hyper-heuristics need to be tested more widely on different problem sets to ascertain their ability to generalise. There are four different benchmark sets that can be used for this purpose [13, 15].

6 Conclusion

This paper aims at providing a foundation on which the domain of hyper-heuristics for the nurse rostering problem can be developed further. The paper firstly presents an overview of the previous work in this domain and then presents a critical analysis of this research to identify directions for future research. The study has revealed that there has not been previous research into the use of selection constructive, generation constructive and generation perturbative hyper-heuristics and thus the evaluation of these hyper-heuristics for solving nurse rostering problems is a potential area for further research. Another area that needs investigating is the derivation of both constructive and perturbation low-level heuristics for the nurse rostering domain. Hyper-heuristics for this domain also need to be tested more widely on a variety of problems. A hybridization of alternately searching a heuristic and solution space appears to be promising and should be explored further. The use of hybrid hyper-heuristics combining for example selection perturbative and selection constructive hyper-heuristics is another area that deserves investigation. These areas for further development of research in this field will form the basis of future work.

Bibliography

[1] Aickelin, U. and Dowsland, K. (2000). Exploiting Problem Structure in a Genetic Algorithm Approach to a Nurse Rostering Problem. Journal of Scheduling. 3 (3): 139-153.

[2] Bai, R., Burke, E. K., Kendall, G., Li, J., McCollum, B. (2010). A Hybrid Approach to the Nurse Rostering Problem. IEEE Transactions on Evolutionary Computation. 14(4): 580-590.

[3] Bilgin, B., De Causmaecker, P., Vanden Berghe, G. (2009). A Hyper-Heuristic Approach to Belgian Nurse Rostering. In proceedings of the Multidisciplinary International Conference on Scheduling: Theory and Applications (MISTA 2009), 683-689.

[4] Bilgin, B., Demeester, P., Misir, M., Vancroonenburg, W., Vanden Berghe, G., Wauters, T. (2010). A Hyper-Heuristic Combined with a Greedy Shuffle Approach to the Nurse Rostering Problem. Online. https://www.kuleuven-kulak.be/˜u0041139/nrpcompetition/abstracts/ l3.pdf [Cited, June 29th, 2012].

[5] Burke, E., De Causmaecker, P., Vanden Berghe, G., Van Landeghem, H. (2004). The State of the Art of Nurse Rostering. Journal of Scheduling. 7: 441-499.

[6] Burke, E., Hart, E., Kendall, G., Newall, J., Ross, P., Schulenburg, S. (2003). Hyper-Heuristics: An Emerging Direction in Modern Research. Handbook of Metaheuristics. Chapter 16: 457-474.

[7] Burke, E. K., Hyde, M., Kendall, G., Ochoa, G., Ozcan, E. (2009). A Survey of Hyper-Heuristics. Computer Science Technical Report No. NOTTCS-TR-SUB- -0906241418-2747. Online. http://www.cs.nott.ac.uk/TR/SUB/SUB-0906241418-2747.pdf . [Cited, June 29th, 2012].

[8] Burke, E. K., Hyde, M., Kendall, G., Ochoa, G., Ozcan, E., Woodward, J. (2009). Exploring Hyper-Heuristic Methodologies with Genetic Programming. Computational Intelligence. 6: 177-201.

[9] Burke, E. K., Hyde, M., Kendall, G., Ochoa, G., Ozcan, E., Woodward, J. (2010). A Classification of Hyper-Heuristic Approaches. Handbook of Metaheuristics. 146: 449-468.

[10] Burke, E. K., Kendall, G., Soubeiga, E. (2003) A Tabu-Search Hyper-Heuristic for Timetabling and Rostering. Journal of Heuristics. 6: 451-470.

[11] Cheng, B., Li, H., Lim, A., Rodrigues, B. (2003). Nurse Rostering Problems - A Bibliographic Survey. European Journal of Operational Research. 151:447-460.

[12] Cowling, P., Kendall, G., Soubeiga, E. (2002). Hyper-Heuristics: A Robust Optimization Method Applied to Nurse Scheduling. PPSN VII, Lecture Notes in Computer Science. 2439: 851-860.

[13] De Causmaecker, P., Vanden Berghe, G. (2011). A Categorization of Nurse Rostering Problems. Journal of Scheduling. 14: 3-16.

[14] De Causmaecker, P., Vanden Berghe, G. (2012). Towards a Reference Model for Timetabling and Rostering. Annals of Operations Research. 194: 167-176.

[15] Haspelagh, S., De Causmaecker, P., Stolevik, M., Schaerf, A. (2012). The First International Competition on Nurse Rostering 2010. Annals of Operations Research. DOI: 10.1007/s10479-012-1062-0.

[16] Ochoa, G., Hyde, M., Curtois, T., Vazquez-Rodriguez, J. A., Walker, J., Gendreau, M., Kendall, G. , McCollum, B., Parkes, A. J., Petrovic, S., Burke, E. K. (2012). HyFlex: A Benchmark Framework for Cross-Domain Heuristic Search. In proceedings of the European Conference on Evolutionary Computation in Combinatorial Optimization (EvoCOP 2012). 7245: 136-147.

[17] Treunicht, M. J., Lane-Visser, T. E., Van Dyk, L., Friedrich, S. S. (2011) A Nurse Rostering Algorithm for a District Hospital in South Africa. In proceedings of the 2011 ORSSA Annual Conference. 93-102.

Proceedings of the 2012 ORSSA Annual Conference
pp. 125–134
www.orssa.org.za/wiki/uploads/Conf/2012ORSSAConferenceProceedings.pdf

ORSSA Proceedings
ISBN
978-07972-1406-4
© 2012

The use of Global Positioning Systems in travel surveys: Experience from a pilot project[1]

JH Nel[2] SC Krygsman[3]

Abstract

Travel surveys which include questions on mode of travel, duration of travel, type of activities and the length of activities for example, are essential requirements for transport planning. Transport modelling incorporates information from origin-destination tables, of which the trip between home and work is seen as most important.

While still a relatively under-researched field, the use of Global Positioning Systems (GPS) to collect travel behaviour data has already made some contributions in the accuracy of trips and activity reporting. It also provides many benefits for respondents such as low input requirements and thus little, if any, respondent burden such as trying to recall activities or trips.

A total of 176 employees at a university took part in the GPS tracking study. They were tracked for two days, and completed a corresponding travel diary as well as a household travel questionnaire.

A method, using cluster analysis, was developed to convert the GPS records of each participant into an activity diary as well as a travel diary. Trip statistics and activity statistics were compared with actual values extracted from the travel diary. Home and work locations were identified and compared to actual home and work.

Results compare favourably with those from the travel diary and are in line with findings in previous unrelated studies. Respondents seem to over report their time spent at work and under report their travel time. Pre- and post-processing of GPS data are cumbersome, but the collection of GPS data is effortless and the cluster analysis technique proved to be useful to extract relevant information.

Key words: GPS technology, travel surveys, cluster analysis, average linkage, activity diary, travel diary.

[1] This study was funded by Royal HaskoningDHV.

[2] Corresponding author: Department of Logistics, Stellenbosch University, Private Bag X1, Matieland, 7602, South Africa, email: jhnel@sun.ac.za.

[3] Department of Logistics, Stellenbosch University, Private Bag X1, Matieland, 7602, South Africa.

1 Introduction

Travel surveys which include questions on mode of travel, duration of travel and type of activities for example, are essential input requirements for transport planning. Wolf [12] reported that the most fundamental and important data are home and work locations of the respondent. Home and work form the majority of the trip-end locations, and knowledge of these locations substantially reduces the effort in activity/trip purpose identification.

While still a relatively under-researched field, the existing literature does reveal that there are potentially many benefits in using Global Positioning Systems (GPS) to collect travel behaviour data. It leads to improved accuracy of trips and activity reporting and better route and speed data [7]. The technology also provides many benefits for respondents, including low input requirements and thus little, if any, respondent burden.

However, few studies report actual comparisons between analysed GPS data and travel surveys with reference to travel time, number of trips, number of activities, the nature of the activities and distance travelled.

The overall objective of this research is to develop an approach to determine the usefulness of GPS in travel and activity surveys. This includes considering the similarity of the GPS derived data and information revealed by the travel questionnaires and trip diaries with respect to the number of trips and activities, travel and activity times, and the identification of home and work locations.

The first step in the process will be to give a brief summary of some literature and techniques used to extract information from the GPS data. Next, the survey will be described, and a detailed description of a method of extracting information from the GPS data will be provided, and some results will be reported.

2 Literature study

A growing body of literature considering the impact of new information communication technologies in travel surveys has emerged. GPS travel surveys, in particular, have become familiar as travel data collection tools. There are surprisingly few studies where GPS and travel surveys have been compared directly, that is, as an 'either or'. Most of the work on comparing GPS results to diaries have been done as part of validation efforts (in particular to assess the accuracy of larger scale travel surveys) or focussed on in-vehicle travel and in-vehicle GPS units [13], [11], [4]. Generally, the results indicate a significant under-reporting of trips in travel. Furthermore, results reveal that the respondents, while under-reporting distance, over-reported their travel times. The attention given to in-vehicle trips has been acceptable to modellers and traffic planners who are focussed on vehicle trips, especially in vehicle dependent USA where most of these GPS studies take place [13].

All the literature reviewed emphasise to some extent the need for rather comprehensive data pre-processing. GPS units record the location at intervals of up to 1 second. Daily tracking (24 hours) at one second intervals can potentially deliver 86 400 records per person! Most GPS units have motion detectors which result in the GPS units not recording position when not in motion. The quantity of information can still be quite significant.

Stopher, Jiang and FitzGerald [10], amongst others, described procedures to identify stops from the GPS navigational streams and breaking in the streams to identify individual trip segments. Periods of non-movement are identified, and if this period exceeds a certain threshold, the presence of a stop is inferred.

Bhat, Srinivasan, Bricka, Ghosh, Sivakumar and Kapur [3] concluded that if a stop duration is longer than 120 seconds, a "stop" is inferred. Axhausen, Schönfelder, Wolf, Oliveria & Samaga [2] classified a trip-end if the dwell time exceeds 5 minutes, a probable trip-end if the dwell time is between 2 and 5 minutes and a suspicious delay if the dwell time is between 20 seconds and 2 minutes.

As noted by Schuessler and Axhausen [7], the appropriate post-processing procedures are still an on-going research issue. While significant progress has been made as Wolf [13] states, the key research questions remain on how to detect individual trips and activities, how to derive the modes used by the participants and how to extract the routes chosen on the network.

3 Pilot survey methodology

Employees at Stellenbosch University were canvassed during June 2010 to take part in a GPS tracking exercise. Staff willing to participate completed consent forms discussing the terms of the study and their right to withdraw at any time. The consent form was a requirement imposed by the Research Ethics Committee of the Stellenbosch University for ethical clearance of this research project.

A total of 176 employees responded. These respondents were from the Bellville/Brackenfell region (14%), Cape Town region (7%), Gordon's Bay/Strand region (10%), Paarl/Wellington region (6%), Somerset West region (11%) and Stellenbosch region (52%). Postgraduate students were trained in administrating the questionnaires and instructed in the use of the GPS devices. The students contacted the respondents to arrange a meeting to deliver the GPS and discuss the questionnaire. The GPS was pre-charged and delivered the day preceding the tracking of the respondents. Respondents were requested to keep the GPS with them for the following two working days of tracking. The tracking days were also noted in the questionnaire. The question "Did you travel to work as you usually travel?" was also asked.

It was decided to set the interval for recording information at 2 minutes or 500 meters. This implies that, at a minimum, a signal was reported every two minutes. While driving or during any movement, the GPS reported locational information every 500 meters, if a distance of more than 500 meters was travelled from the previous location before the end of the two minute interval. This arrangement ensured a continuous stream of location information.

In order to extend battery life, the units were set to automatically turn off when no motion was detected. Respondents were also requested to charge the units at night.

In addition to the GPS, respondents were asked to complete a travel diary and a household travel questionnaire.

4 Extracting information from the GPS data

While the GPS units deliver exact results of spatial locations, it does so only through the recording and delivering of enormous amounts of data. A unit set to record a location every two minutes delivers approximately 720 records per day. Units can be programmed to go to sleep when motionless which does reduce the number of records quite significantly. It is not uncommon to receive between 500 and 2000 records per day per person (depending on the interval setting between location data). Extraction of usable information from this raw data can be a cumbersome and complex task.

The GPS devices delivered raw data as shown in Table 1. Each respondent was tracked for two weekdays, and each day for each respondent was treated as a separate diary. The information was imported into ArcGIS [1] and X and Y coordinates (in meters) were added (last two columns in Table 1). The GPS data were imported in SAS [6]. An approach to identify trips and activities was developed and the rules and heuristics were programmed in SAS to extract the information. The next paragraphs describe the process in detail.

Cluster analysis, specifically the average linkage method [9], was used to compute hierarchical clusters of the (x; y) coordinates. These clusters are viewed as potential stopping locations. This process eliminates the use of an extensive number of heuristics to decide on issues such as potential activity start or potential activity ends, for example, as described by Scheussler and Axhausen [7], Stopher *et al.* [10] and others.

Table 1: *Raw GPS data*

ID	GPSID	DATE_	TIME_	Longitude	Latitude	Altitude	Location	X_coor	Y_coor
Ant001	11070000576215	2010/11/04	08:02:11	18.8534	-33.9018	184	KYLEMORE, STELLENBOSCH, MOUNT SILVER	-13557.6	-3752776.8
Ant001	11070000576215	2010/11/04	08:03:11	18.8571	-33.9019	182	KYLEMORE, STELLENBOSCH, HENDRIKSE	-13217.0	-3752787.4
Ant001	11070000576215	2010/11/04	08:04:11	18.8586	-33.9017	180	KYLEMORE, STELLENBOSCH, HENDRIKSE	-13075.2	-3752770.6
Ant001	11070000576215	2010/11/04	08:05:11	18.8596	-33.9077	192	KYLEMORE, IDAS VALLEY, ADAM TAS STREET	-12986.4	-3753426.7
Ant001	11070000576215	2010/11/04	08:05:37	18.8591	-33.9134	186	KYLEMORE, STELLENBOSCH, ADAM TAS STREET	-13027.1	-3754062.7
Ant001	11070000576215	2010/11/04	08:05:54	18.8584	-33.9170	182	KYLEMORE, STELLENBOSCH, ADAM TAS STREET	-13092.9	-3754460.3
Ant001	11070000576215	2010/11/04	08:06:54	18.8609	-33.9219	164	KYLEMORE, STELLENBOSCH, HELSHOOGTE ROAD	-12860.9	-3755001.6
Ant001	11070000576215	2010/11/04	08:08:07	18.8709	-33.9238	164	KYLEMORE, STELLENBOSCH, HELSHOOGTE ROAD	-11940.6	-3755213.0
Ant001	11070000576215	2010/11/04	08:08:07	18.8709	-33.9238	164	KYLEMORE, STELLENBOSCH, HELSHOOGTE ROAD	-11940.6	-3755213.0
Ant001	11070000576215	2010/11/04	08:08:43	18.8774	-33.9252	162	KYLEMORE, IDAS VALLEY, HELSHOOGTE ROAD	-11339.4	-3755373.1
Ant001	11070000576215	2010/11/04	08:09:43	18.8775	-33.9253	161	KYLEMORE, IDAS VALLEY, HELSHOOGTE ROAD	-11324.0	-3755376.8
Ant001	11070000576215	2010/11/04	08:10:50	18.8745	-33.9298	144	KYLEMORE, STELLENBOSCH, CLUVER	-11602.3	-3755880.0
Ant001	11070000576215	2010/11/04	08:11:44	18.8713	-33.9336	125	KYLEMORE, STELLENBOSCH, VICTORIA STREET	-11896.1	-3756303.7
Ant001	11070000576215	2010/11/04	08:12:43	18.8670	-33.9336	124	KYLEMORE, STELLENBOSCH, VICTORIA STREET	-12296.7	-3756300.5
Ant001	11070000576215	2010/11/04	08:13:08	18.8669	-33.9335	121	KYLEMORE, STELLENBOSCH, DE BEER	-12307.5	-3756291.3
Ant001	11070000576215	2010/11/04	08:13:44	18.8676	-33.9336	124	KYLEMORE, STELLENBOSCH, VICTORIA STREET	-12238.2	-3756302.3

In average linkage the distance between two clusters is viewed as the average distance between pairs of observations, one in each cluster. Average linkage tends to join clusters with small variances, and it is slightly biased toward producing clusters with the same variance [6]. It also provides a centroid for each cluster, which is essential for this application. The centroid method and Ward's minimum variance method were also investigated, but the centroid method in general does not perform as well as the average linkage method, and Ward's method is strongly biased toward producing clusters with roughly the same number of observations, which is not applicable in this study [6].

The cluster analysis was repeated specifying 25, 50, 75 and 100 clusters on each dataset, i.e. for each subject and each day separately. The purpose of selecting 4 different sets of clusters for each dataset was to assess the accuracy of results obtained by the clustering analysis. It was noticed, by inspection, that there is a positive relationship between the size of the activity space (the product of the [difference between the maximum and minimum y coordinates] and the [difference between the maximum and minimum x coordinates]), and the number of clusters that should be used to correctly identify a specific location.

Figure 1 shows the outcome of cluster analysis when 25 and 100 clusters were used respectively, on one of the datasets. The impact of the cluster size is clearly visible in the number of (x; y) locations added to a specific cluster.

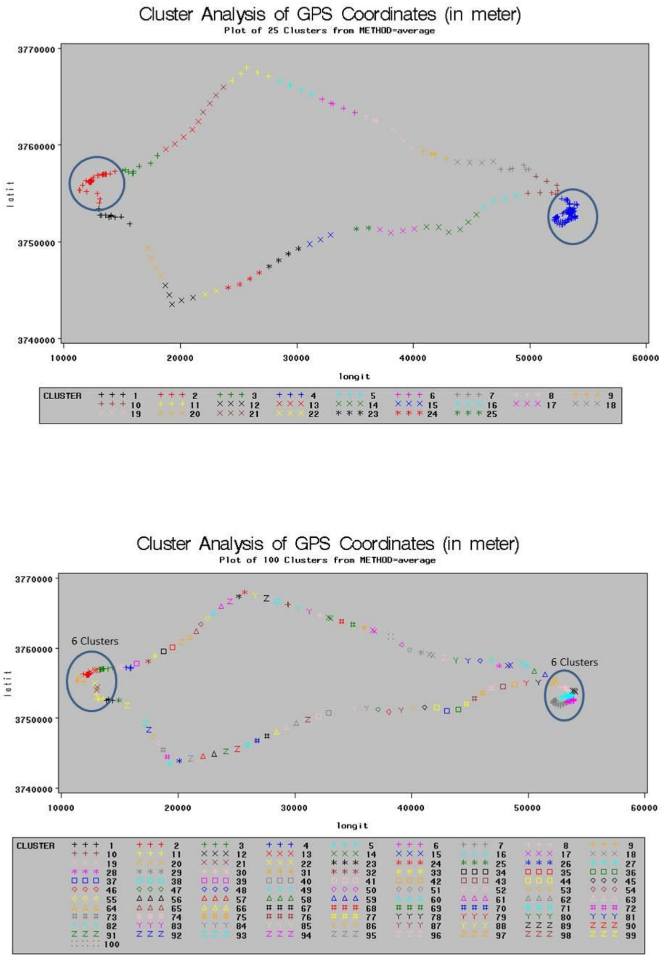


Figure 1: *Cluster Analysis*

The fewer the clusters, the more (x; y) coordinates are included in each cluster. Too few clusters might result in parts of the trip to be included in the cluster, too many clusters might result in representing a stop with different allocations. The process of selecting an appropriate cluster size will be described at the end of the paragraph, and will relate to the size of the activity space.

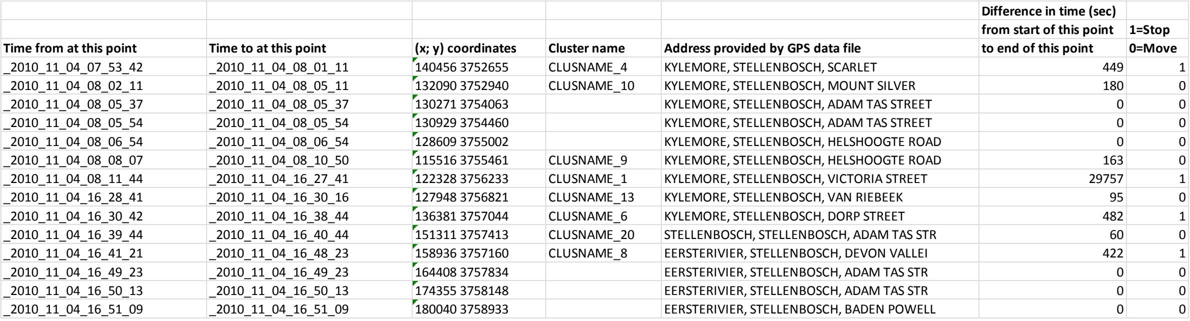
After the cluster analysis was performed only the clusters with the largest number of (x; y) coordinates allocated to them were retained. The first 20 clusters were used (this can be changed using a heuristic and the assumption is that not more than 20 stops occurred). An $(\bar{x};\bar{y})$ coordinate, representing the centroid of the cluster, was computed using SAS [6], to represent each of these clusters. All the (x; y) locations falling in these clusters were recoded to have the same $(\bar{x};\bar{y})$ coordinate.

The (x; y) locations in the remaining clusters (other than the first 20 clusters) kept their original coordinate values. There was generally a noticeable decline in the number of coordinates allocated to a cluster after the first 3 – 4 clusters. As clusters are potential locations, the clustering procedure provided an indication of the number of activity locations visited.

The following step involved an aggregation of the records based on the newly defined (x; y) locations and the time at each (x; y) location. Records were sorted by time, and successive records were aggregated into groups if their newly defined (x; y) coordinates were the same. During this process, the times corresponding with the first and the last (x; y) coordinates in the group were recorded and seen as "time from" and "time end" of the group. If the coordinate corresponds to one which represents one of the renamed clusters, the name of the cluster is also recorded. The process is repeated for the next group of coordinates, until the end of file is reached. Table 2 shows part of a file created after the process of data aggregation.

A variable "*Stop*" is added. If the duration exceeds 300 seconds, as suggested by Axhausen *et al.* [2], a stop ("*Stop*"=1) is identified.

Table 2: *Data Aggregation Process*

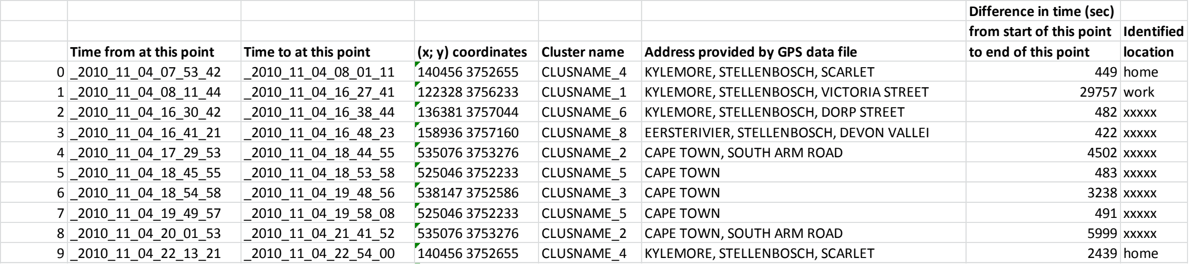


					Difference in time (sec)	
					from start of this point	1=Stop
Time from at this point	**Time to at this point**	**(x; y) coordinates**	**Cluster name**	**Address provided by GPS data file**	**to end of this point**	**0=Move**
_2010_11_04_07_53_42	_2010_11_04_08_01_11	140456 3752655	CLUSNAME_4	KYLEMORE, STELLENBOSCH, SCARLET	449	1
_2010_11_04_08_02_11	_2010_11_04_08_05_11	132090 3752940	CLUSNAME_10	KYLEMORE, STELLENBOSCH, MOUNT SILVER	180	0
_2010_11_04_08_05_37	_2010_11_04_08_05_37	130271 3754063		KYLEMORE, STELLENBOSCH, ADAM TAS STREET	0	0
_2010_11_04_08_05_54	_2010_11_04_08_05_54	130929 3754460		KYLEMORE, STELLENBOSCH, ADAM TAS STREET	0	0
_2010_11_04_08_06_54	_2010_11_04_08_06_54	128609 3755002		KYLEMORE, STELLENBOSCH, HELSHOOGTE ROAD	0	0
_2010_11_04_08_08_07	_2010_11_04_08_10_50	115516 3755461	CLUSNAME_9	KYLEMORE, STELLENBOSCH, HELSHOOGTE ROAD	163	0
_2010_11_04_08_11_44	_2010_11_04_16_27_41	122328 3756233	CLUSNAME_1	KYLEMORE, STELLENBOSCH, VICTORIA STREET	29757	1
_2010_11_04_16_28_41	_2010_11_04_16_30_16	127948 3756821	CLUSNAME_13	KYLEMORE, STELLENBOSCH, VAN RIEBEEK	95	0
_2010_11_04_16_30_42	_2010_11_04_16_38_44	136381 3757044	CLUSNAME_6	KYLEMORE, STELLENBOSCH, DORP STREET	482	1
_2010_11_04_16_39_44	_2010_11_04_16_40_44	151311 3757413	CLUSNAME_20	STELLENBOSCH, STELLENBOSCH, ADAM TAS STR	60	0
_2010_11_04_16_41_21	_2010_11_04_16_48_23	158936 3757160	CLUSNAME_8	EERSTERIVIER, STELLENBOSCH, DEVON VALLEI	422	1
_2010_11_04_16_49_23	_2010_11_04_16_49_23	164408 3757834		EERSTERIVIER, STELLENBOSCH, ADAM TAS STR	0	0
_2010_11_04_16_50_13	_2010_11_04_16_50_13	174355 3758148		EERSTERIVIER, STELLENBOSCH, ADAM TAS STR	0	0
_2010_11_04_16_51_09	_2010_11_04_16_51_09	180040 3758933		EERSTERIVIER, STELLENBOSCH, BADEN POWELL	0	0

The following step was the extraction of activities and the trips separately with associated attribute information.

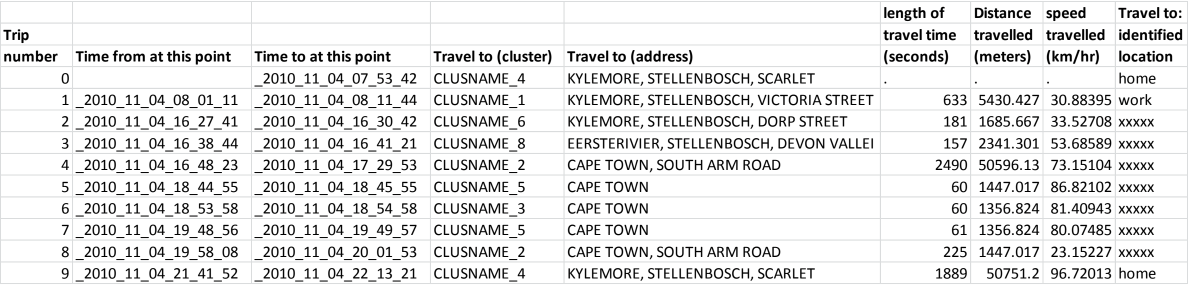
The aggregated data in Table 2 was converted to activities by keeping all the records with "*Stop*"=1 and calculating the attributes as described next. Table 3 shows the extracted activity data and Table 4 the extracted travel diary. The following attributes are also listed: *length of travel to next stop*, *length of stay at this stop, distance (m) to next stop, speed travelled* and *identified location.*

Table 3: *Extracted activity diary*



						Difference in time (sec)	
						from start of this point	Identified
	Time from at this point	**Time to at this point**	**(x; y) coordinates**	**Cluster name**	**Address provided by GPS data file**	**to end of this point**	**location**
0	_2010_11_04_07_53_42	_2010_11_04_08_01_11	140456 3752655	CLUSNAME_4	KYLEMORE, STELLENBOSCH, SCARLET	449	home
1	_2010_11_04_08_11_44	_2010_11_04_16_27_41	122328 3756233	CLUSNAME_1	KYLEMORE, STELLENBOSCH, VICTORIA STREET	29757	work
2	_2010_11_04_16_30_42	_2010_11_04_16_38_44	136381 3757044	CLUSNAME_6	KYLEMORE, STELLENBOSCH, DORP STREET	482	xxxxx
3	_2010_11_04_16_41_21	_2010_11_04_16_48_23	158936 3757160	CLUSNAME_8	EERSTERIVIER, STELLENBOSCH, DEVON VALLEI	422	xxxxx
4	_2010_11_04_17_29_53	_2010_11_04_18_44_55	535076 3753276	CLUSNAME_2	CAPE TOWN, SOUTH ARM ROAD	4502	xxxxx
5	_2010_11_04_18_45_55	_2010_11_04_18_53_58	525046 3752233	CLUSNAME_5	CAPE TOWN	483	xxxxx
6	_2010_11_04_18_54_58	_2010_11_04_19_48_56	538147 3752586	CLUSNAME_3	CAPE TOWN	3238	xxxxx
7	_2010_11_04_19_49_57	_2010_11_04_19_58_08	525046 3752233	CLUSNAME_5	CAPE TOWN	491	xxxxx
8	_2010_11_04_20_01_53	_2010_11_04_21_41_52	535076 3753276	CLUSNAME_2	CAPE TOWN, SOUTH ARM ROAD	5999	xxxxx
9	_2010_11_04_22_13_21	_2010_11_04_22_54_00	140456 3752655	CLUSNAME_4	KYLEMORE, STELLENBOSCH, SCARLET	2439	home

Table 4: *Extracted travel diary*



					length of	Distance	speed	Travel to:
Trip					travel time	travelled	travelled	identified
number	Time from at this point	Time to at this point	Travel to (cluster)	Travel to (address)	(seconds)	(meters)	(km/hr)	location
0		_2010_11_04_07_53_42	CLUSNAME_4	KYLEMORE, STELLENBOSCH, SCARLET	.	.	.	home
1	_2010_11_04_08_01_11	_2010_11_04_08_11_44	CLUSNAME_1	KYLEMORE, STELLENBOSCH, VICTORIA STREET	633	5430.427	30.88395	work
2	_2010_11_04_16_27_41	_2010_11_04_16_30_42	CLUSNAME_6	KYLEMORE, STELLENBOSCH, DORP STREET	181	1685.667	33.52708	xxxxx
3	_2010_11_04_16_38_44	_2010_11_04_16_41_21	CLUSNAME_8	EERSTERIVIER, STELLENBOSCH, DEVON VALLEI	157	2341.301	53.68589	xxxxx
4	_2010_11_04_16_48_23	_2010_11_04_17_29_53	CLUSNAME_2	CAPE TOWN, SOUTH ARM ROAD	2490	50596.13	73.15104	xxxxx
5	_2010_11_04_18_44_55	_2010_11_04_18_45_55	CLUSNAME_5	CAPE TOWN	60	1447.017	86.82102	xxxxx
6	_2010_11_04_18_53_58	_2010_11_04_18_54_58	CLUSNAME_3	CAPE TOWN	60	1356.824	81.40943	xxxxx
7	_2010_11_04_19_48_56	_2010_11_04_19_49_57	CLUSNAME_5	CAPE TOWN	61	1356.824	80.07485	xxxxx
8	_2010_11_04_19_58_08	_2010_11_04_20_01_53	CLUSNAME_2	CAPE TOWN, SOUTH ARM ROAD	225	1447.017	23.15227	xxxxx
9	_2010_11_04_21_41_52	_2010_11_04_22_13_21	CLUSNAME_4	KYLEMORE, STELLENBOSCH, SCARLET	1889	50751.2	96.72013	home

The *identified location* needs to be clarified. If the names of the first cluster and the last cluster in the activity diary are the same, that cluster, representing a specific location, is identified as "*home*". All activities in the diary at that location are renamed to "*home*". If the first and last records are not represented by the same cluster, both are recorded as "*home?*", and other activities in the diary at the same locations as both "*home?*" are renamed. The "*work*" decision is based on the duration (longest stay) at an activity other than the activities already labelled as "*home*" or "*home?*". All other activities (stays) not classified as "*home*", "*home?*" or "*work*" were identified as "*Other Activities*", labelled as "xxxxx".

Lastly, the activity spaces of all the subjects, for different days separately, were calculated. The 25th, 50th and 75th percentiles of the activity spaces were calculated. Those diaries with activity spaces smaller than the 25th percentiles were modelled using 25 clusters, 50 clusters for activity spaces between the 25th and 50th percentiles, 75 clusters for activity spaces between the 50th and 75th percentiles and 100 clusters for those activity spaces above the 75th percentile. One way to assess the validity of these diaries is to consider the speed of each trip within the trip diary. If the minimum speed is less than 0.5 km/hour or the maximum speed greater than 160 km/hour, it indicates that the set of GPS data contains records that need pre-processing. This can be due to an inadequate number of satellites required for accurate positioning, for example. If movement takes place almost immediately after ignition-on, it may take some seconds for signal acquisition. This is one of the major reasons why "*home*" is not uniquely identified. Pre-processing of data is necessary.

5 Comparison

The GPS records were easy to collect, as opposed to the completed travel diary, where the reliability of the results depends on the respondent to report all trips as accurately as possible. On the other hand, the pre- and post-processing of the GPS diaries are cumbersome and post-processing is still an ongoing issue. The advantage, however, of the accuracy of the results that can possibly be obtained by using the GPS as compared to the travel diary, makes the GPS more appealing.

The following tables compare the GPS trip and activity diaries, and the completed travel diaries (which was also converted to an activity diary). There were 176 subjects (352 days), but only 315 diaries (from the GPS data and completed travel diary) were completed. Information regarding trips and activities *of all* subjects are compared on the left hand side of Table 5. The right hand side of Table 5 provides information regarding trips and activities of those subjects, *excluding* the 43 diaries for which the GPS data should have been pre-processed (13.7%). These 43 diaries were identified by the fact that there were instances where the calculated travelling speed was out of range. More than half of these cases were instances where people travelled short distances, and the activity space, seen as a rectangle, was less than 33km^2.

In this study, the GPS records (excluding diaries which needed pre-processing) one more trip (median value for data excluding problem cases) compared to the travel diary. The travel times seem longer for the GPS records, but the extra trip should be considered. The travel according to the GPS records also includes all stroll trips, i.e. walk trips at work locations or at other activity locations[4], which might have been excluded in the completed travel diary. Interestingly, the GPS compares relatively well with the diary in terms of total activity time (excluding home), 8.6 hours as opposed to 8.7 hours. This time is made up of time at work (6.6 hours obtained for GPS, and 7.4 hours reported in the diaries) and at other locations (1.7 hours obtained for GPS, and 1.1 hours for travel diary). It seems that while users over-estimate their time at their formal work location, they tend to under-estimate their time at other locations.

Table 5: *Comparing results: GPS and travel (trip and activity) diaries*

	All subjects, day 1 and day 2 viewed as separate datasets				**Data excluding problem cases, day 1 and day 2 viewed as separate datasets**		
	TRIPS (n=315)	**Diary (Mean, Median, q1 – q3)**	**GPS (Mean, Median, q1 – q3)**		**TRIPS (n=275)**	**Diary (Mean, Median, q1 – q3)**	**GPS (Mean, Median, q1 – q3)**
1	Number of trips	5.9 5.0 4 - 8	9.3 7.0 4 – 12		Number of trips	6.0 6.0 4 - 8	8.7 7.0 4 - 11
2	Travel time (min)	87.8 80.0 50 - 115	163.9 115.4 80 - 186		Travel time (min)	89.9 82.0 52 - 118	146.4 105.6 77 - 152
3	Distance travelled (km)		47.8 41.3 14 – 63		Distance travelled (km)		49.4 42.8 16 - 64
	ACTIVITIES (n=312) (Mean, Median, q1 – q3)	**Diary**	**GPS**		**ACTIVITIES (n=272) (Mean, Median, q1 – q3)**	**Diary**	**GPS**
4	Number activities (excl home)	4.5 4.0 3 - 6	6.9 5.0 3 – 8		Number activities (excl home)	4.6 4.0 3 - 6	6.5 5.0 3 – 8
5	Number activities (excl home and work)	2.7 2.0 1 - 4	4.4 3.0 1 – 5		Number activities (excl home and work)	2.8 2.0 1 - 4	4.1 3.0 1 – 5
6	Total activity time (hrs) (excl home)	8.4 8.7 7.4 – 9.6	8.5 8.5 6.8 – 9.9		Total activity time (hrs) (excl home)	8.4 8.7 7.3 – 9.6	8.5 8.6 7.2 – 9.8
7	Total activity time (hrs) (excl home and work)	1.7 1.0 0.2 – 2.6	2.3 1.7 0.5 – 3.4		Total activity time (hrs) (excl home and work) (n=202)	1.7 1.1 0.3 – 2.8	2.3 1.7 0.5 – 3.2
	ACTIVITIES AT WORK ONLY (n=300) (Mean, Median, q1 – q3)	**Diary**	**GPS**		**ACTIVITIES AT WORK ONLY (n=263) (Mean, Median, q1 – q3)**	**Diary**	**GPS**
8	Number of activities (work only)	1.9 2.0 1 - 2	2.4 1.0 1 – 2.5		Number of activities (work only)	1.9 2.0 1 - 2	2.3 1.0 1 – 2
9	Total activity time at work only (hrs)	7.0 7.5 5.6 – 8.6	6.2 6.5 4.3 – 8.2		Total activity time at work only (hrs)	6.9 7.4 5.6 – 8.5	6.3 6.6 4.3 – 8.2

These results also correspond with results obtained by other authors [5], [13]. Wolf [13] noted that under-reporting of trips was found to be around 16%. Approximately 3 activities are spent on other than home or work. This includes mostly activities such as shopping on the way home, or picking up children at school. This information can be extracted from the travel diaries. The GPS diaries contain information on home and work activities only. Work locations were reasonable correctly identified in approximately 73% of the cases, by comparing street addresses. The most important reasons for the 27% not being identified correctly were that the work address could not be identified exactly from the completed travel diary (9%) or that the respondents spent more time during the day at locations other than work (16%). There were 57 cases recorded as "did not travel to work as usually travelled", but this did not appear to be the

[4] The respondents were asked to keep the GPS with them at all times.

reason for not being able to identify the work location correctly. About 75% of the cases identified the home location correctly. Again, the main reason for the 25% not being identified correctly is that the home location could not be identified exactly from the completed travel diary (7%) and in 18% of the cases a completely different venue was identified.

6 Conclusions and recommendations

This study was a first attempt to use cluster analysis to identify possible stops, recode the coordinates and aggregate the GPS records into sequences of trips and stops to eventually compile activity and trip diaries. Several heuristics were used. First of all, the average linkage method was used; a stop was defined to be longer than 300 seconds; valid speeds were taken as between 0.5 km/hour and 160 km/hour; and the number of clusters used was a function of the size of the activity space, which was calculated as a rectangle. Even with this first attempt, acceptable results were achieved. About 86% of the diaries were used, and the need for pre-processing of the GPS data was highlighted. It was also noted that in cases where the activity spaces were very small, especially for people who travelled less than 5 km to work, the clustering procedures delivered poorer results as in the cases where respondents travelled further. This should be investigated further.

This study shows that it is possible to identify home and work locations with reasonable accuracy. It is possible to derive meaningful travel data using GPS, with clustering methods.

Bibliography

[1] ArcGIS 10.1. Esri. New York.

[2] Axhausen, K.W., Schönfelder, S., Wolf, J., Oliveria, M. and Samaga, U. (2004). Eighty weeks of GPS traces: Approaches to enriching trip information. Transportation Research Board 83rd Annual Meeting Pre-print CD-ROM.

[3] Bhat, C.R., Srinivasan, S., Bricka, S., Ghosh, P., Sivakumar, A. and Kapur, A. (2006). Conversion of volunteer-collected GPS diary data into travel time performance measures. Summary Report 0-5176-S, prepared for the Texas Department of Transportation, February.

[4] Forrest, T. and Pearson, D. (2005). Comparison of trip determination methods in household travel surveys enhanced by a Global Positioning System. *Transportation Research Record: Journal of the Transportation Research Board*, No. 1917, 63-71.

[5] London Area Travel Survey. (2003). Department of Transport.

[6] SAS Institute Inc. (2002). SAS 9.1.3. Cary, NC. USA.

[7] Schuessler, N. and Axhausen, K.W. (2008). Identifying trips and activities and their characteristics from GPS raw data without further information. Paper presented at the 8th International Conference on Survey Methods in Transport. May 25-31, Annecy, France.

[8] Sharp, J. and Murakami, E. (2004). Travel survey methods and technologies: Resource Paper. Presented at the National Household Travel Survey Conference: Understanding our Nation's Travel, Washington D.C.

[9] Sokal, R.R. and Michener, C.D. (1958). A statistical method for evaluating systematic relationships. University of Kansas Science Bulletin. 38:1409 – 1438.

[10] Stopher, P., Jiang, Q. and FitzGerald, C. (2005). Processing GPS data from travel surveys. Paper presented to the Second International Colloquium on the Behavioural Foundations of

Integrated Land-Use and Transportation Models: Frameworks, Models and Applications, Toronto, June.

[11] Stopher, P., FitzGerald, C. and Xu, M. (2007). Assessing the accuracy of the Sydney household travel survey with GPS. *Transportation.* 34:723-741.

[12] Wolf, J. (2000). Using GPS data loggers to replace travel diaries in the collection of travel data, Dissertation, Georgia Institute of Technology, Atlanta.

[13] Wolf, J. (2006). Applications of new technologies in travel surveys. In Stopher, P.R., Stecher, C.C. (eds.) *Travel Survey Methods—Standards and Future Directions*, pp. 531–544. Elsevier, Oxford.