A multi-resolution approach towards point-based multi-objective geospatial facility location

Andries M. Heyns \textsuperscript{a,b,*}, Jan H. van Vuuren \textsuperscript{b}

\textsuperscript{a} Department of Logistics, Stellenbosch University, South Africa
\textsuperscript{b} Department of Industrial Engineering, Stellenbosch University, South Africa

\begin{abstract}

The placement of certain facilities, such as radars and wind turbines, requires careful planning according to very specific geographical and spatial requirements. Such placement problems are often solved by metaheuristics which find near-optimal solutions within a fraction of the time required to solve these problems exactly. The use of high-resolution representations of the feasible search space generally ensures a high level of solution quality and accuracy, but involves evaluation of a larger number of candidate solutions than lower resolution representations, and is therefore more time-consuming. A trade-off between solution quality and time requirements must therefore be achieved when choosing an appropriate resolution of data to include in geospatial facility location models. In this paper, we propose a novel explore-and-exploit, multi-resolution solution approach that takes advantage of the reduced computational requirements associated with lower resolution terrain data, while simultaneously benefiting from the quality of solutions returned at higher resolutions. Our multi-resolution approach is capable of outperforming analyses in which only highest resolution data are considered, both in terms of solution quality and solution time requirements.

\end{abstract}

1. Introduction

Research into the optimal placement of facilities according to geographical and spatial criteria — henceforth referred to as geospatial facility location problems (GFLoPs) — are well-documented and wide-ranging in solution methodology and practical application. A large portion of GFLoPs are suitability analyses that are region-based and aim to find generally large, contiguous areas of terrain destined for the placement of a number of facilities within their borders, e.g., regions identified for the development of wind farms (Sliz-Szkliniarz & Vogt, 2011; Van Haaren & Fthenakis, 2011) or solar farms (Sanchez-Lozano, Teruel-Solano, Soto-Elvira, & Socorro Garcia-Cascales, 2013; Uyan, 2013). This paper, however, is concerned with point-based problems in which the aim is to find precise, discrete facility site locations for networks of facilities which generally include one type of facility, e.g., watchtowers (Agarwal et al., 2005), transmitters (Akella, Delmelle, Batta, Rogerson, & Blatt, 2010; Krzanowski & Raper, 1999; Lee & Murray, 2010), surveillance sensors (Bao, Xiao, Lai, Zhang, & Kim, 2015; Kim, Murray, & Xiao, 2008; Murray, Kim, Davis, Machiraju, & Parent, 2007) and wind turbines (Emami & Noghreh, 2010; Kwong et al., 2014; Serrano-González, Burgos-Payán, & González-Longatt, 2013). Point-based analyses may often follow ones that are region-based.

The space of location decisions in point-based facility location problems is generally categorised as continuous or discrete (ReVelle & Eiselt, 2005). In continuous problems, the points to be sited can generally be placed anywhere on the plane, while in discrete problems the facilities can be placed only at a limited number of pre-selected candidate sites (eligible points) on the plane. We solve GFLoPs as discrete facility location problems in this paper — for which raster data are used to provide the pre-selected candidate sites. Raster data represent the earth’s surface and environmental information as uniformly spaced sample points, called gridposts, across the terrain surface. Gridposts that lie within feasible facility placement regions, such as those identified in region-based analyses, may be considered for facility site placement and are called candidate sites. Raster data are employed extensively in the literature for solving point-based GFLoPs due to its ease of implementation — examples include the placement of wind turbines (Kwong et al., 2014; Serrano-González et al., 2013), radar and weapon systems (Chose, Prasad, & Guruprasad, 1993; Taner Güçlü, Maras, Gencer, & Aygunes, 2013), and other line-of-sight (LOS)-dependent facilities (Franklin, 2002; Kim, Rana, & Wise, 2004; Heyns & Van Vuuren, 2015a; Lee & Murray, 2010).

A natural approach towards solving a point-based GFLoP is to select a single resolution of geospatial data to include in the model, after which a search algorithm may be employed to find suitable candidate site combinations based on these data. Higher resolution data include more candidate sites spaced closely together, whereas lower resolution data include fewer candidate locations spaced further apart. The use of

* Corresponding author at: Department of Logistics, Stellenbosch University, South Africa.

E-mail address: 13554379@sun.ac.za (A.M. Heyns).

http://dx.doi.org/10.1016/j.compenvurbys.2016.01.007
0198-9715 © 2016 Elsevier Ltd. All rights reserved.
high-resolution data therefore involves a larger number of candidate sites and associated evaluations than lower resolution representations to arrive at (near-)optimal solutions and are therefore more time-consuming, but this generally ensures a high level of solution quality and accuracy. Lower resolution representations of the data may be extracted from high-resolution data with the aim of reducing the number of candidate sites and the computational complexity associated with solving the problem — ultimately resulting in shorter computation times. This, however, typically comes at a loss of solution quality due to the potential loss of good candidate sites from the higher resolution data. A trade-off between solution quality and computation time requirements must therefore be achieved when choosing an appropriate resolution of terrain data to use in geospatial facility location models.

In this paper we present a new multi-resolution approach (MRA) towards reducing the computational burden of solving the problem by reducing the number of candidate sites that are evaluated during the optimisation process, while the superior solution quality typically associated with higher resolution analyses is maintained. In fact, the solution quality of the new approach is consistently superior to that of the traditional single-resolution approach (SRA) in which only the highest resolution data are considered. Both approaches followed in this paper employ the popular Non-dominated Sorting Genetic Algorithm-II (NSGA-II) (Deb, Pratap, Agarwal, & Meyarivan, 2002) to search for solution alternatives for a visibility-related implementation of the bi-objective backup coverage location problem (BCLP) (Hogan & ReVelle, 1986; Kim et al., 2008; Murray et al., 2007).

The paper opens with a discussion on important concepts and background information related to the work presented. Descriptions of the SRA and MRA solution approaches towards solving raster-based GFLoPs follow in Section 3.1 and Section 3.2, respectively. A scenario involving a visibility-related BCLP is introduced in Section 4 for the purpose of illustrating the two approaches and comparing their results. The paper closes with a brief conclusion and proposals for future work in Section 5.

2. Background

An illustration of a raster data representation of terrain is provided in Fig. 1(a). The section of terrain surface shown in this figure is, in fact, a graphical representation of sampled elevation data at the gridposts (the empty grey dots). Search zones (SZs) are feasible facility placement regions specified on the terrain surface and envelop the candidate sites that may be considered for facility placement (the solid dots). A candidate solution is a specific configuration of a number of facilities (three for the example in the figure) at candidate sites in the SZ. Depending on the type of facilities and criteria considered for the placement problem, candidate sites may be evaluated with respect to gridposts enveloped within specified interest zones (IZs), called interest points (the black squares). A significant portion of GFLoPs that involve IZs are visibility-related and require LOS analyses (Agarwal et al., 2005; Goodchild & Lee, 1989; Heyns & Van Vuuren, 2013,2015a; Kim et al., 2004; Lee, 1991; Nagy, 1994; Tabik, Zapata, & Romero, 2013; Zhao, Padmanabhan, & Wang, 2013).

The notions of solution domination and of a Pareto-front of non-dominated solutions in objective function space are illustrated in Fig. 2. Crossover and mutation operations performed on candidate solutions are shown in Fig. 2.
The BCLP — an extension to the maximal covering location problem — is a member of the category of covering location problems. Covering problems are thoroughly investigated and applied in the literature (Bao et al., 2015; Goodchild & Lee, 1989; Kim et al., 2008, 2004; Lee & Murray, 2010; Owen & Daskin, 1998; ReVelle & Eiselt, 2005; Tanergüçü et al., 2013). In the context of this paper, the BCLP includes the objectives of (1) maximising primary coverage, and (2) maximising mutual (overlapping) coverage of a network of LOS-dependent facilities with respect to an IZ. Other problem categories that may also be included in GFLoPs include p-dispersion and p-centre problems (Erdut & Neuman, 1991; Hakimi, 1965; Owen & Daskin, 1998), which may be implemented in studies that consider criteria such as distance from features, and solar energy — additional examples of such GFLoPs solved by the approaches put forward in this paper are available (Heyns & Van Vuuren, 2015b).

The process of configuring a candidate solution takes place in the solution space, as illustrated in Fig. 1(a), and such a candidate site combination corresponds to a single point in objective function space which measures the solution’s scores with respect to the placement criteria, as illustrated in Fig. 1(b). Since the objectives of facility location problems are often conflicting in nature, a set of well-performing trade-off solutions is sought. More specifically, non-dominated solutions are sought, while their dominated counterparts are avoided. Non-dominated solutions are at least as good as those that are dominated with respect to all the objectives, and better than the dominated solutions with respect to at least one (Knowles, Thiele, & Zitzler, 2006). Collectively, the non-dominated solutions form a so-called Pareto-front in the objective function space, which is sought by decision makers in order to choose a solution that seems most appropriate — both in terms of the solution’s proposed facility site locations on the physical terrain and its performance values with respect to the placement criteria. The notion of solution dominance is illustrated in Fig. 1(b) for a (dominated) candidate solution such as the one in Fig. 1(a), along with the notion of a Pareto-front.

Solving multi-objective (MO) problems in pursuit of the exact Pareto-front is often associated with a substantial computational burden which may become prohibitively large (Kim et al., 2008; Murray et al., 2007; Owen & Daskin, 1998; ReVelle & Eiselt, 2005). Powerful metaheuristic optimisation procedures are often employed in such instances in order to approximate the set of solutions in the Pareto-front. One approach that is often followed is the weighted-sum method, in which solutions are the result of a weighted combination of criteria-related performance values — examples in a GFLoP context include wind turbine micro-siting (Emami & Noghrehi, 2010; Kwong et al., 2014; Serrano-González et al., 2013) and the BCLP (Hogan & ReVelle, 1986; Murray et al., 2007). By varying the objective weights in multiple approximation runs, a Pareto-front approximation may be ‘traced’ out. This approach has, however, been shown to hold numerous disadvantages, such as the laborious and sensitive iterative process of assigning suitable weights to the objectives (Das & Dennis, 1997; Stanimirović, Zlatanović, & Petković, 2011), the possibility of obtaining misleading and/or biased results (Stewart, 2007), and the requirement of multiple approximation runs in order to trace out the Pareto-front approximately (Das & Dennis, 1997; Hughes, 2005; Kim et al., 2008). MO evolutionary algorithms,1 on the other hand, can approximate the Pareto-front in a single run and, in the context of a visibility-related BCLP, it has been demonstrated by Kim et al. (2008) that such an MO approach may offer substantial benefits over the weighted-sum method used by Murray et al. (2007). For the abovementioned reasons we adopt an MO optimisation approach in this paper. The NSGA-II (Deb et al., 2002) is the MO evolutionary algorithm employed in this paper — our implementation’s ability to identify all the non-dominated solutions in the exact Pareto-front, but within drastically reduced computation times compared to an exhaustive search, has been demonstrated (Heyns & Van Vuuren, 2015b).2

Data resolution is not only important in respect of complexity concerns as previously discussed in Section 1. The practical implementation of solutions in terms of site placement certainty — which is typically related to maximum displacement distance from candidate sites (Murray, 2003; Yeh & Chow, 1996) — suggests that proposed solution sites should preferably be analysed and presented at the highest possible resolution — as long as this is within computational reach. Finally, computation time and data storage requirements are reduced if the number of candidate sites that are evaluated in order to approximate the Pareto-front is minimised. This may be achieved by reducing the number of candidate sites with the use of lower resolution data representations — which is in conflict with the preference of high-resolution data for site placement certainty. This conflict of resolution preference is resolved by the MRA.

3. Solution approaches

The processes followed to solve GFLoPs according to the SRA and MRA are discussed in Sections 3.1 and 3.2, respectively.

3.1. Single-resolution search approach

In this section we only provide a brief description of our implementation of the NSGA-II for solving a GFLoP at a single resolution — a thorough description of the development of the algorithm for similar purposes, including a pseudocode description of the algorithm, is provided by Kim et al. (2008) and an example of its practical implementation in the context of GFLoPs is described by Heyns and Van Vuuren (2015a).

The general process of the NSGA-II starts with the stochastic generation of an initial population of candidate site solution combinations of size N according to a uniform distribution. The NSGA-II then iteratively generates offspring populations — each offspring population is the result of carefully controlled evolutionary operators applied to a ‘matting pool’ of solutions that are selected stochastically from the previous population. This process is repeated until some termination criterion is met — in this paper this was when new generations no longer improved significantly over previous generations in terms of their performance with respect to the objective functions.

For a facility location scenario involving the placement of three facilities, a candidate solution is represented as a chromosome (string) of three candidate facility site numbers. The facility site numbers are pre-determined by an indexing scheme, typically derived with

---

1 The interested reader is referred to Deb et al. (2002), Fonseca and Fleming (1993), Purshouse and Fleming (2003) and Zitzler et al. (2000) for more detailed descriptions of the working of evolutionary algorithms.

2 A similar NSGA-II implementation utilised by Kim et al. (2008) also indentified all the non-dominated solutions in their problem.
respect to row and column indices. Suppose, for example, that two parent solutions have been selected for modification, with the first solution comprising facility sites 12, 10 and 13, while the second solution comprises facility sites 4, 6 and 15, as shown in the first parts of Figs. 2(a) and 3(a).

A tournament is employed to select two parents from the mating pool which proceed to undergo modification in the form of crossover and mutation operations. Each set of parent solutions results in two offspring solutions. A specific point along the chromosome is chosen, the two strings are cut at this point, and the heads and tails of these strings are exchanged. This type of crossover results in new site combinations as solutions, but does not alter the constituent sites of solutions. The resulting offspring solutions may be seen in the second parts of Figs. 2(a) and 3(a) with respect to solution space and their string representations, respectively. Since the parents selected for crossover typically perform well with respect to the objective functions, the offspring solutions inherit some of the strong properties of their parents, but at the same time also explore new solution combinations. Not all parent solutions selected from the mating pool undergo crossover. Instead, crossover is subject to a limited number of offspring solutions. Not all parent solutions selected from the mating pool proceed to undergo modification, with the first solution comprising facility sites 4, 6 and 15, as shown in the second parts of Figs. 2(b) and 3(b). Mutation occurs for each solution in the post-crossover offspring population with a mutation probability, denoted by $p_m$. Mutation is introduced after the crossover stage and promotes solution diversity in the sense of introducing new sites into the site combinations of solutions, as opposed to merely exchanging these sites from an existing collection of sites. This is achieved by arbitrarily selecting a site in the chromosome and exchanging it for a site that is randomly selected from the search space, as illustrated in Figs. 2(b) and 3(b). Mutation occurs for each solution in the post-crossover offspring population with a mutation probability, denoted by $p_m$. The offspring solutions and the previous population are combined, from which the $N$ best solutions are selected to form the new population.

3.2. A novel multi-resolution search approach

Observations made during experimental studies using the SRA in conjunction with the NSGA-II at different data resolution levels for specific problem instances — which led us to the development of the novel MRA — are discussed in this section.

3.2.1. Motivation

During multiple runs of the SRA for identical problem instances, the sites included in the solutions making up the final populations and approximate Pareto-fronts are often observed to be from the same regions. Analogously, poorly performing areas, which often envelop large numbers of candidate sites, are typically absent from the final sets of solutions and these absent regions typically exhibit much larger sizes than the areas covered by those containing final solution sites. If the initial population is generated stochastically over the entire SZ, candidate sites that eventually do not form part of the final population of solutions are therefore needlessly analysed. Accumulatively, consideration of these sites can be very costly in terms of the computation times required to calculate their performance values with respect to the objectives. For example, the computation of a single candidate site’s viewshed in respect of all the points in a 10 km × 10 km IZ, using a data resolution of approximately 28 m between gridposts, takes on average 3.7 s.

An example of a typical collection of all the sites evaluated during a hypothetical run of the SRA is provided in Fig. 4(a), in addition to the final sites that are included in the non-dominated solutions. It is clear that a large number of candidate sites are evaluated which are not included in the final solutions. For statistical and empirical reliability, it is standard practice to solve the problem through multiple runs of a stochastic metaheuristic (Knowles et al., 2006). As may be seen in Fig. 4(b), multiple runs result in even more weak candidate sites being evaluated, while the collection of sites that are included in the non-dominated solution combinations uncovered by multiple runs continue to appear in the same well-performing regions.

Improved search efficiency may be achieved if the search for solutions can be focussed towards the candidate sites in the well-performing areas, while avoiding the remaining sites. One way of achieving this is to perform criteria-specific analyses in order to identify candidate sites that are likely to be included in well-performing candidate solutions. Visibility-related objectives are considered in this paper and an example of a visibility-related identification of likely placement sites is provided by Kim et al. (2004) — here, specific terrain analyses are performed to identify areas on the terrain surface that are likely to achieve good visibility of the surrounding terrain for the purposes of subsequent viewshed-dependent facility location. The inclusion of objectives that are not related to visibility (e.g., distance from features) would, however, render such a procedure meaningless, since the filtering of areas that perform well with respect to visibility-related objectives has no relation to distance-related criteria. Moreover,
Fig. 5. The extraction of lower resolution candidate sites.

(a) Maximum resolution sites  
(b) Sites extracted at lower resolutions

Fig. 6. Sites included in approximately Pareto-optimal solutions determined during low-resolution analyses generally occur in the vicinity of those obtained during high resolutions, but may be calculated in a fraction of the corresponding time. Low-resolution sites may therefore be used to quickly identify well-performing regions that merit further exploration.

(a) Sites included in Pareto-optimal solution combinations of high and low-resolution analyses  
(b) Corresponding approximate Pareto-fronts

Fig. 7. An illustration of the MRA solution process.
it may take a large amount of time to perform the necessary analyses for the identification of potentially good placement areas. This type of criteria-specific filtering is therefore not considered in our research.

An alternative approach towards reducing the number of candidate sites with the aim of reducing computational complexity is to perform resolution-based site extraction. Candidate sites at the highest available data resolution may simply be reduced to uniformly spaced lower resolution subsets, as illustrated in Fig. 5. The SRA may then be applied to find candidate solution combinations from the candidate sites at the lower resolutions. Typical results of the different resolution data analyses in solution and objective function spaces are provided in Fig. 6(a) and (b). As may be seen in Fig. 6(b), low-resolution Pareto-fronts are observed to be inferior to their high-resolution counterparts due to ‘good’ candidate sites being overlooked for inclusion in candidate solutions. The inferior fronts may, however, be determined in a significantly shorter amount of time, due to fewer candidate sites being considered and smaller population sizes being required. As illustrated in Fig. 6(a), low-resolution solution sites are often observed to correspond with — or at least be in the vicinity of — some of those returned by higher resolution analyses. These recurring observations led us to the process of first exploring lower resolution data to quickly identify well-performing regions, after which higher resolution data may be exploited within the vicinity of these regions in order to improve on solution quality.

3.2.2. Method

The MRA starts by accepting as input the SZ at the lowest resolution, after which it enters a loop during which the problem instance is repeatedly solved (approximately) at gradually increasing resolutions (exploitation). All the sites that are included in the solutions of the final population — or only those from the approximated Pareto-fronts, depending on user preference — are carried over to the next resolution. The problem is then solved for a new SZ which includes the carried-over solution sites and a selection of their higher-resolution neighbours (exploitation). The steps in this loop are repeated until the problem instance is solved at the maximum resolution. Fig. 7 provides a graphical description of this solution process.7

3.2.3. Important considerations

The choice of which resolution levels to include in the model — in terms of the number of levels and each level’s spacing distances — may be a parameter pre-specified by the user, or it may be computed dynamically during the solution process. The quality of the final solutions is highly dependent on the locations of the solution sites identified during analyses performed at the first (lowest) resolution level, since subsequent analyses at higher resolutions will be limited to candidate sites in their vicinity. If this resolution level is chosen too low, there is a risk of the solution space being overly simplified, resulting in potentially well-performing candidate sites and their sub-regions being overlooked and lost. It is therefore critical to select the initial resolution level so that it contains sufficient information for the identification of candidate sites that lie within well-performing sub-regions. An appropriate initial resolution level is expected to depend primarily on terrain roughness, features and size — complex, ‘rugged’ terrains are expected to require higher initial resolution levels than terrain that is undulating.

The range of neighbourhood exploitation when moving into higher resolution levels is another important consideration. In this paper a simple \( n \times n \) site span, centred around each candidate site carried over, is used to introduce new candidate sites when moving into higher resolutions.2 If the range of neighbourhood exploitation is too small, there is a risk of reducing the solution space too quickly, resulting in weak solution diversity and potentially weak solution quality due to the premature loss of good candidate sites. If this range is too large, however, more candidate sites, including more weaker ones, may be introduced into the population — resulting in increased computation times and potentially weaker solution quality.

4. Worked example

This section is devoted to an illustration of the working of the SRA and MRA in the context of solving a realistic BCLP instance for the purposes of comparing solution quality and computation time.

4.1. Terrain description

A section of terrain with gentle hills and multiple peaks was chosen in an area of the South African Western Cape, surrounding the town of Malmesbury. The terrain is square and measures 10 km south to north

---

6 It is important to note that a lower resolution representation merely considers fewer candidate sites within the SZ on the terrain surface, while the analysis data computed with respect to the LZ and objective function criteria are always determined using the highest resolution data.

7 A pseudocode description of the MRA is provided in supporting documentation (Heyns & Van Vuuren, 2015b).

8 Selecting new candidate sites according to a neighbourhood radius instead of a site span is another approach that may be considered. Dynamic neighbourhood site spans or radii, determined according to each resolution level, are alternative approaches that may be considered.
and 10 km west to east. Fig. 8(a) contains a relief representation of the terrain elevation. The highest available resolution of terrain data, namely 1 arc sec or approximately 28 m spacing between gridposts, was obtained, resulting in a total of 125,125 candidate sites.

4.2. Problem description

Suppose that three facilities with visibility/detection requirements (e.g., watchtowers, radars or telecommunication towers) are to be placed on the terrain in Fig. 8 and that they have the following characteristics:

- The facilities have visible distances that extend past the boundaries of the terrain. Viewsheds are therefore determined with respect to the entire surface area.
- The height of the facilities are such that the viewpoint is 5 m above the ground. This is important for viewshed computation since the elevated point of view results in improved terrain visibility.

Suppose the entire terrain surface may be considered for facility placement. All gridposts in the terrain are therefore candidate sites included in the SZ. The visibility requirements of the entire terrain surface is considered equally important and so all the gridposts are points in the IZ.

Suppose the first objective is to place the three facilities in such a manner that the number of interest points which are visible from at least one of the three facilities is maximised. The purpose of this objective is to ensure maximum coverage of the IZ. Furthermore, suppose another objective is to place the three facilities in such a manner that the number of interest points which are virtually visible from at least two of the three facilities is maximised. The purpose of this objective is to ensure that, in the event of a facility being unmanned or not operational, the area of interest affected by the loss of visibility from the particular facility is minimised (backup cover).

Suppose the placement is subject to the two constraints that exactly three facilities are to be placed, and that at most one facility can be placed at any candidate site.

4.3. Mathematical problem formulation

The mathematical formulation for the example follows the BCLP formulation suggested by Hogan and ReVelle (1966) and implemented for surveillance sensor placement purposes by Murray et al. (2007) and Kim et al. (2008) — we exclude importance values in respect of the IZ gridposts.

Denote the set of candidate sites within the SZ by $S$, define the binary decision variable

$$x_s = \begin{cases} 1, & \text{if a facility is placed at candidate site } s \in S, \\ 0, & \text{otherwise} \end{cases}$$

and denote the set of gridposts within the IZ by $I$. We define the auxiliary variables

$$y_i = \begin{cases} 1, & \text{if gridpost } i \in I \text{ is visible from at least one facility located in } S, \\ 0, & \text{otherwise} \end{cases}$$

and

$$u_i = \begin{cases} 1, & \text{if gridpost } i \in I \text{ is visible from at least two facilities located in } S, \\ 0, & \text{otherwise} \end{cases}$$

The height of the facilities are such that the viewpoint is 5 m above the ground. This is important for viewshed computation since the elevated point of view results in improved terrain visibility.

Furthermore, the parameters

$$v_{i,s} = \begin{cases} 1, & \text{if gridpost } i \in I \text{ is visible from candidate site } s \in S, \\ 0, & \text{otherwise} \end{cases}$$

are computed a priori or during each optimisation run. The objectives are then to

$$\text{maximise } V_1 = \sum_{i \in I} y_i$$

subject to the constraints

$$\sum_{s \in S} x_s - y_i - u_i \geq 0, \quad i \in I,$$

$$u_i - y_i \leq 0, \quad i \in I,$$

$$\sum_{s \in S} x_s = 3,$$

$$x_s \in \{0, 1\}, \quad s \in S,$$

$$y_i, u_i \in \{0, 1\}, \quad i \in I.$$

Furthermore, the parameters are computed a priori or during each optimisation run. The objectives are then to

$$\text{maximise } V_1 = \sum_{i \in I} y_i$$

subject to the constraints

$$\sum_{s \in S} x_s - y_i - u_i \geq 0, \quad i \in I,$$

$$u_i - y_i \leq 0, \quad i \in I,$$

$$\sum_{s \in S} x_s = 3,$$

$$x_s \in \{0, 1\}, \quad s \in S,$$

$$y_i, u_i \in \{0, 1\}, \quad i \in I.$$

### Table 1

Values of NSGA-II parameters, discussed in Section 3.1, selected for the SRA and MRA in the context of solving the problem instance of Section 4.

<table>
<thead>
<tr>
<th></th>
<th>$N$</th>
<th>$p_f$</th>
<th>$p_m$</th>
</tr>
</thead>
<tbody>
<tr>
<td>SRA</td>
<td>1800</td>
<td>0.8</td>
<td>0.15</td>
</tr>
<tr>
<td>MRA (12)</td>
<td>200</td>
<td>0.8</td>
<td>0.125</td>
</tr>
<tr>
<td>MRA (6, 3, 1)</td>
<td>$\sqrt{n}/5$</td>
<td>0.8</td>
<td>0.125</td>
</tr>
</tbody>
</table>

---

The choice of solving the problem for three facilities is only for illustrative purposes. Examples with larger numbers of facilities to be placed are provided in supporting documentation (Heyns & Van Vuuren, 2015b).
4.4. Parameter selection

In order to use the NSGA-II to solve Eqs. (1)–(7), a number of parameters have to be specified, as discussed in Section 3.1. The aim is to assign values to the parameters that result in an approximate Pareto-front achieving a good spread of solutions and good solution quality — these values should therefore be chosen carefully and is a difficult, problem instance-specific, empirical process.

The highest resolution of terrain data available, including 125,125 candidate sites in the SZ, was used for solving the problem according to the SRA. A detailed sensitivity analysis was performed to arrive at the set of suitable SRA parameter values in Table 1. The population size of 1800 is unusually large and the reason for this is that the aim was to find parameter combinations that consistently returned the best solution quality and spread of solutions in order to compare the best results returned by the SRA to those of the MRA. The fact that the highest resolution terrain data were used for the SRA further necessitated large population sizes in order to sufficiently explore the solution space. Population sizes smaller than 1800 returned approximate Pareto-fronts of weaker and inconsistent quality, albeit within faster computation times. The unusually high mutation probability after crossover compensates for the lack of inherent site mutation of the fixed-point crossover.

Table 2

<table>
<thead>
<tr>
<th>Mean (std. deviation)</th>
<th>Minimum</th>
<th>Maximum</th>
<th>Attainment front</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>SRA</strong></td>
<td>0.852 (0.081)</td>
<td>0.688</td>
<td>0.949</td>
</tr>
<tr>
<td><strong>MRA</strong></td>
<td>0.972 (0.015)</td>
<td>0.936</td>
<td>0.989</td>
</tr>
</tbody>
</table>

The highest resolution of terrain data available, including 125,125 candidate sites in the SZ, was used for solving the problem according to the SRA. A detailed sensitivity analysis was performed to arrive at the set of suitable SRA parameter values in Table 1. The population size of 1800 is unusually large and the reason for this is that the aim was to find parameter combinations that consistently returned the best solution quality and spread of solutions in order to compare the best results returned by the SRA to those of the MRA. The fact that the highest resolution terrain data were used for the SRA further necessitated large population sizes in order to sufficiently explore the solution space. Population sizes smaller than 1800 returned approximate Pareto-fronts of weaker and inconsistent quality, albeit within faster computation times. The unusually high mutation probability after crossover compensates for the lack of inherent site mutation of the fixed-point crossover.

Fig. 10. Twenty Pareto-front approximations of (a) the SRA and (b) the MRA, and (c) the corresponding attainment fronts obtained by solving the problem instance of Sections 4.1–4.3 according to the two solution approaches. In (d), the attainment fronts obtained by the MRA at its separate resolution levels are shown.

10 Examples with more generally used SRA population sizes are provided in supporting documentation (Heyns & Van Vuuren, 2015b).
operator used in this paper—more generally used binary representations of decision variables employ crossover-points chosen randomly along binary strings (De Jong & Spears, 1992), resulting in the crossover operator also affecting mutation.

Four MRA resolution levels were chosen for the problem instance considered in this paper, following an empirical testing process. The starting resolution level was chosen as 12 arc sec spacing, which roughly translates to 340 m between candidate sites (as opposed to the maximum available resolution of 1 arc sec, or approximately 28 m between candidate sites). This resolution contains 924 candidate sites (as opposed to 125,125 at the maximum resolution), which are illustrated in Fig. 8(b). After the initial resolution level, the subsequent levels were chosen as 6, 3 and the maximum level of 1 arc sec. When entering a new resolution level, the new candidate sites were selected as a 5×5 site span centred around each candidate site carried over.

With respect to the MRA parameter selection, a sensitivity analysis similar to that carried out in respect of the SRA was performed to arrive at a parameter combination that returned the best solution quality and spread of solutions for the initial resolution level, and these values are provided in Table 1. Compared to the SRA, this parameter set only differs in population size — due to the significantly smaller number of candidate sites — and a marginally smaller mutation rate. In selecting the best population size for the resolution levels following the initial level, it was generally found that population sizes of a fifth of the number of carried-over and newly introduced candidate sites returned consistently good results — in addition to keeping the remaining parameter values of the initial resolution. The solution sites carried over to each new resolution level were selected as all the sites included in the final population, as opposed to those in the approximately Pareto-optimal solutions only. This approach returned better solutions, mostly with respect to the overall spread in objective function space as a result of the greater level of site diversity.

4.5. Performance evaluation and comparison

We now present a summary of the results uncovered by twenty separate runs of each of the SRA and MRA in the context of the scenario described in Sections 4.1–4.3 for the parameter sets motivated in Section 4.4. 11

4.5.1. Pareto-fronts

Two of the main aims in MO optimisation are (i) to produce an approximate Pareto-front at a minimised distance from the true Pareto-front, assuming its location is known, and (ii) to maximise the smoothness and uniformity of the distribution of the Pareto approximation set (Zitzler, Deb, & Thiele, 2000). A unary indicator which measures the extent to which this has been achieved is the hypervolume measure (Knowles et al., 2006). It measures the total hypervolume in the objective function space that is dominated by a Pareto approximation set relative to a reference point which is dominated by the entire solution set. An example of the hypervolume measure is provided in Fig. 9. More space is dominated by larger hypervolumes, hence these are desired. For the purposes of this paper, a graphical comparison of results in the form of the attainment sets of algorithms (the globally best set of non-dominated solutions from all optimisation runs) and the hypervolume measure are used to evaluate algorithmic performance.

The optimisation results are presented graphically in objective function space (with total versus mutual visibility expressed as a percentage of all the gridposts included in the terrain) in Fig. 10, first in the form of the twenty separate Pareto-fronts and the attainment front for (a) the SRA, and (b) the MRA. In Fig. 10(c), the two attainment fronts of the respective approaches are compared — all SRA solutions are dominated by solutions in the MRA front. The progression of the approximate Pareto-fronts obtained at the different resolution levels of the MRA is illustrated in Fig. 10(d) in the form of the attainment fronts of the approximated sets at each level. The values of the hypervolume measure (normalised by the hypervolume of the attainment front of the MRA) are shown in Table 2 for the two approaches. The hypervolumes were calculated with respect to a reference point at 45% for objective 1 and 26% for objective 2 — the origin in Fig. 10(a) and (b).

From the fronts in Fig. 10 and the normalised hypervolume values in Table 2 it may be seen that the MRA approximate Pareto-fronts clearly outperform those of the SRA. The MRA achieves a small standard deviation of the hypervolume indicator, which corresponds to the fronts in Fig. 10(b) lying closely together. Similarly, the larger standard deviation of the hypervolume indicator for the SRA corresponds to the fronts in Fig. 10(a) being more scattered. The minimum hypervolume value of the MRA is only marginally smaller than the maximum value for the SRA, and is not much smaller than the corresponding value of the SRA attainment front. Based on these values, it is not only anticipated that any run of the MRA will be superior to an SRA run, but also that a single MRA run will return an approximation set that is close to the final MRA attainment set — resulting in fewer MRA runs required to obtain a good attainment set.

Fig. 11 contains the physical site locations and visibility-related results for the two solutions in the MRA attainment front of Fig. 10(c) that return the highest value with respect to each of the objectives. These figures encapsulate the type of information that would typically

---

11 Another option would be to consider smaller population sizes, but with much larger mutation rates — that would result in slower computation times and require more generations for convergence. In either instance, the size of the search space necessitates unusually large parameter values for adequate solution space exploration.

12 Graphical illustrations of the candidate site locations at the four resolution levels are available in supporting documentation (Heyns & Van Vuuren, 2015b).
be provided to decision makers in order to facilitate a final decision with respect to the placement of facilities.

4.5.2. Sites evaluated and computation times

This section is devoted to a comparison of the number of sites evaluated by the SRA and MRA, as well as a comparison of their resulting computation times. It is important to note that each of the twenty runs for each respective approach commenced with no pre-computed viewshed data. Instead, a caching approach was followed, whereby only the viewsheds of sites included in candidate solutions evaluated during the solution processes were computed and stored to disc. These viewsheds were recalled in the event that a site was again included in candidate solutions of subsequent analyses — hence avoiding the costly re-computation of viewshed data.

The number of sites evaluated and the corresponding computation times of both approaches are presented visually in Fig. 12, employing dual-scale vertical axes in Fig. 12(a)–(c) which represent the number of sites evaluated in addition to computation times.

The data presented in Fig. 12(a) are of specific relevance to GFLoPs that require resource-intensive analyses, such as viewshed

---

**Table 3**

A comparison of average population sizes, numbers of generations and metaheuristic computation times between the SRA and the four resolution levels of the MRA.

<table>
<thead>
<tr>
<th></th>
<th>Population size (std. deviation)</th>
<th>Generations (std. deviation)</th>
<th>Metaheuristic computation time (std. deviation)</th>
</tr>
</thead>
<tbody>
<tr>
<td>SRA</td>
<td>1800</td>
<td>72 (28)</td>
<td>29 m 48 s (11 m 24 s)</td>
</tr>
<tr>
<td>MRA (12 arc sec)</td>
<td>200</td>
<td>58 (22)</td>
<td>1 m 8 s (23 s)</td>
</tr>
<tr>
<td>MRA (6)</td>
<td>174 (33)</td>
<td>57 (30)</td>
<td>1 m 2 s (38 s)</td>
</tr>
<tr>
<td>MRA (3)</td>
<td>200 (37)</td>
<td>57 (25)</td>
<td>1 m 16 s (36 s)</td>
</tr>
<tr>
<td>MRA (1)</td>
<td>242 (58)</td>
<td>60 (20)</td>
<td>1 m 46 s (40 s)</td>
</tr>
</tbody>
</table>
computations. This is because the results of such analyses are typically stored in either random access memory or on disc for each site evaluated—so as to be available during subsequent queries. It may be seen that the SRA initially introduces a substantial number of previously unevaluated candidate sites per run since no viewsheds were pre-computed, after which this number decreases as the solution space is explored, until it is ultimately comparable to that of the MRA. The fact that the MRA introduces considerably fewer sites per run — and a notably smaller initial number of sites compared to that of the SRA — results in a considerably smaller computation time for the smaller number of viewsheds. It may be seen, for both approaches, that the computation time per run strongly correlates with the number of sites evaluated, as expected. Since a viewshed took, on average, 3.7 s to compute per candidate site, the total viewshed computation time is a significant contribution to the total computation time per run.

In Fig. 12(b) it may be seen how the SRA accumulatively evaluates almost all of the 125,125 candidate sites (122,729 to be precise) after twenty runs, while the MRA evaluates only 10,187 sites. This dramatic difference is even more significant when considering the fact that the MRA returns superior Pareto-front approximations — a clear indication of its superior search efficiency. The accumulated computation time after each run is related to the accumulated number of sites evaluated, resulting in a total computation time of 5 days and 16 h for the twenty runs of the SRA, while the twenty runs of the MRA took only 12 h and 22 min.\(^\text{13}\)

Fig. 12(c) and (d) are of particular significance to GFLoPs which do not require resource-intensive analyses (such as the computation of viewsheds), or GFLoPs for which such analyses are already pre-computed. The graphs show the computation time of the two approaches when excluding viewshed computation and therefore present the time spent on metaheuristic analysis only. For the NSGA-II this entails initial population generation, offspring generation, solution evaluation, ranking and sorting, and the computation of crowding distances. In Fig. 12(c), it may be seen that the SRA evaluates a considerably larger number of sites per run than does the MRA (a mean of 22,206 compared to 2943), since it requires larger population sizes than those of the MRA in order to explore the terrain surface adequately. This results in a considerable number of population-based operations to be performed, in addition to the algorithm requiring a larger number of generations to converge because of the large solution space to be explored. A direct result of this slow convergence is more generations and an increase in the number of sites evaluated — especially with the large mutation rate of 0.15 used for the SRA. This explains the correlation between the computation times and the number of sites evaluated per run, since the SRA computation times were proportionally related to the number of generations required per run to converge. The average number of generations, as well as the average metaheuristic computation time for the SRA is presented in Table 3. In comparison, the MRA requires smaller population sizes and fewer population-based operations at each resolution level due to the reduced and concentrated search spaces and, as a result, converges faster, as may be seen for the different resolution levels in Table 3. The MRA significantly out-performs the SRA when analysing accumulated (metaheuristic) computation times over twenty runs in Fig. 12(d) — 1 h and 44 min, compared to 9 h and 56 min.

Fig. 13 shows all the sites evaluated during a single run of the SRA (run number 4 with a total of 22,504 sites evaluated, similar to the mean number per run), compared to the 10,187 sites evaluated during all twenty runs of the MRA. It may be seen for the MRA how certain ‘weaker’ areas are evaluated at lower resolutions only and identified as such, while some areas are specifically identified and analysed more intensely at higher resolutions. The progression of candidate sites and resulting final population sites (around which the candidate sites of each new resolution level were centred) of all twenty runs of the MRA at the four resolution levels is shown in Fig. 14, so as to illustrate the manner in which the explore-and-exploit approach seeks out the sites that make up the best solutions.

5. Conclusion and future work

In this paper, two approaches were considered for solving a visibility-related instance of the bi-objective BCLP, incorporating a popular genetic algorithm, the NSGA-II. It was shown that twenty runs of the newly developed MRA exhibited superior performance compared to the same number of runs of the SRA (in which only the highest available resolution data were considered) — both in terms of average and

---

13 The computation times reported here were, in fact, greatly reduced by running up to eight instances of MATLAB in parallel on separate cores. This resulted in total computation times of under one day for the SRA and under two hours for the MRA. Total (serial) computation times are, however, presented in order to better illustrate the correlation between computation time and the number of sites evaluated.
attainment hypervolumes, and the associated computation time. The MRA is able to identify potentially well-performing areas in the SZ and avoids the unnecessary evaluation of poorer areas. As a result, fewer resource-intensive viewshed computations are required, resulting in shorter computation times. The MRA also exhibits shorter metaheuristic computation times when the effects of viewshed computation times are excluded — results provided in supporting documentation (Heyns & Van Vuuren, 2015b) confirmed the expectation that the MRA will typically outperform the SRA for problem instances with access to pre-computed analysis data or for which the analyses are very simple. The supporting documentation also contains descriptions of other GFLoP instances in which the MRA was shown to be effective and superior to the SRA in the context of ‘rough’ terrain, terrain that covers very large areas, and for problems including objectives that are not visibility-related.

The work presented in this paper is in early development and there remain numerous questions that require formal investigation. The selection of resolution levels to include in the model and the neighbourhood exploitation approach, discussed in Section 3.2.3, are examples of these. Other MO metaheuristics may be tested for inclusion in the MRA framework — a swap-based algorithm and a hybrid NSGA-II/swap-based algorithm have, for instance, produced very good results. The use of exact solution methods instead of metaheuristics may also be considered when the number of candidate solutions is sufficiently small. The computation times reported in this paper were serial (accumulated) computation times — in most instances these times were, however, greatly reduced by distributing the workload over parallel cores. Serial computation times were reported in order to provide more meaningful and precise comparisons of the SRA and MRA. Formal research into more efficient implementations of parallelisation and distributed computing alternatives should, however, be investigated, since the work presented in this paper is ideally suited for these alternatives.

Raster data were used in the development of the MRA due to the simplicity it offers in terms of data representation structure and resolution level extraction. GFLoPs in which candidate sites are spaced irregularly, such as is often encountered in indoor planning problems (Lee, 2015), (Zhou, Dao, Thill, & Delmelle, 2015), may be solved by the MRA if it is possible to identify site neighbourhoods according to spatial distributions. The number of sites in each neighbourhood may be reduced to different levels and the problem solved in a manner similar to that followed in the MRA. This approach may be suited for a problem such as that considered by Lee (2015) due to the sites being spaced with relative uniformity, as opposed to the problem considered by Zhou et al. (2015) for which the MRA may be less appropriate. The MRA also offers the opportunity to improve on previous results in which very low resolutions of candidate sites were used (Lee and Murray, 2010). By increasing the resolution around the final solution sites and solving the problem afresh by the MRA, an improved level of site placement certainty and solution quality may be achieved.

Acknowledgements

The financial assistance of the South African National Research Foundation (NRF) (grants 84634 and 97736) and the Armaments Corporation of South Africa (ARMSCOR) towards this research is hereby acknowledged. Opinions expressed and conclusions arrived at, are those of the authors and are not necessarily to be attributed to ARMSCOR or the NRF. Lastly, the authors would like to thank the anonymous reviewers for their constructive input during the revision process of this paper.

Appendix A. Supplementary data

Supplementary data to this article can be found online at http://dx.doi.org/10.1016/j.compenvurbsys.2016.01.007.

References
