International Journal of Geographical Information Science
Vol. 19, No. 5, May 2005, 581–601

Taylor & Francis
Taylor & Francis Group

Research Article

Integration of genetic algorithms and GIS for optimal location search

XIA LI† and ANTHONY GAR-ON YEH‡
†School of Geography and Planning, Sun Yat-sen University, 135 West Xingang Rd., Guangzhou 510275, P.R. China; e-mail: lixia@graduate.hku.hk; gplx@zsu.edu.cn
‡Centre of Urban Planning and Environmental Management, The University of Hong Kong, Pokfulam Road, Hong Kong SAR, P.R. China; e-mail: hdxugoy@hkucc.hku.hk

(*Received 24 February 2004; accepted 23 September 2004*)

Optimal location search is frequently required in many urban applications for siting one or more facilities. However, the search may become very complex when it involves multiple sites, various constraints and multiple-objectives. The exhaustive blind (brute-force) search with high-dimensional spatial data is infeasible in solving optimization problems because of a huge combinatorial solution space. Intelligent search algorithms can help to improve the performance of spatial search. This study will demonstrate that genetic algorithms can be used with Geographical Information systems (GIS) to effectively solve the spatial decision problems for optimally sitting *n* sites of a facility. Detailed population and transportation data from GIS are used to facilitate the calculation of fitness functions. Multiple planning objectives are also incorporated in the GA program. Experiments indicate that the proposed method has much better performance than simulated annealing and GIS neighborhood search methods. The GA method is very convenient in finding the solution with the highest utility value.

Keywords: Genetic algorithms; GIS; Optimal location; Multiple objectives; Simulated annealing

1. Introduction

An often encountered spatial decision problem is to search for the best site or sites to accommodate one or more facilities to generate the best utility values (e.g. the maximum population coverage and minimum transport cost). Traditional location-allocation methods before GIS data were available only use relatively small datasets (Church 1999). The general facility location problem and its variants, including most location-allocation and *p*-median problems, are known to be NP-hard combinatorial optimization problems. Most of these traditional methods cannot easily handle thousands of demand points and sites in GIS datasets (Church 1999). This is especially a problem when raster data with many cells are used. Some types of data aggregation have been used in dealing with large data sets. Although agglomerative clustering can be used to find good solutions, this problem can be complicated by including the items that need to be stored at each location. In this case, clustering will not yield an optimal solution. Goodchild (1979) has pointed out that data aggregation can have a great effect in the absolute location of a specific facility. Consequently, there is now substantial literature on heuristic algorithms for a variety of location problems, among which can be found the well-known simulated annealing algorithm (Simha *et al*., 2001).

International Journal of Geographical Information Science
ISSN 1365-8816 print/ISSN 1362-3087 online © 2005 Taylor & Francis Group Ltd
http://www.tandf.co.uk/journals
DOI: 10.1080/13658810500032388

Location search usually requires the use of optimization tools. There are two categories of optimization methods. The first is the local optimization method, such as simplex, Gauss-Newton, and the Levenberg-Marquart (Zhan *et al.* 2003). These local optimization algorithms have limitations because the search may be trapped in local minima or maxima and their success is heavily dependent on the choice of initial values. The second is the global optimization method which can avoid such problems. These algorithms include simulated annealing and genetic algorithms (GAs). Studies have indicated that GAs are attractive global search tools suitable for the multimodal objective functions (Zhan *et al.* 2003). GAs have advantages for global optimization without using complicated calculations, and they are effective especially when the number of parameters is very large (Jin and Wang 2001). GAs are stochastic search algorithms for searching optimal solutions in large and complex non-linear spaces.

GAs are inspired by Darwin's theory of evolution as a part of evolutionary computing (Rechenberg 1973, Holland 1992). The algorithms adopt an evolutionary process to solve optimization problems based on the mechanism of natural selection. They have been proven excellent in quickly finding solutions to complex optimization problems (Goldberg 1989, Mitchell 1996) and have been successfully applied to a variety of disciplines. Goldberg (1989) demonstrated the flexibility of GAs as the optimization mechanisms are independent of fitness functions. This flexibility is desirable as it enables the modification of evaluation functions without altering the algorithms themselves.

The objective of this paper is to explore the capability of GAs in solving high-dimensional optimization problems in continuous space. Although GAs have been widely used for searching optimal parameter values, there are limited studies on the integration of GAs and GIS for solving optimization problems in resource and environmental management. The computation for solving spatial decision problems is very intensive and conventional search algorithms have difficulties in coping with the complex situations. There are many mathematical methods which can find optimization solutions very quickly for fairly "well-behaved" problems. However, these traditional methods tend to break down when the problem is not so "well-behaved". The use of evolutionary algorithms should be very efficient in solving a lot of spatial decision problems. The integration of GAs and GIS can help to find optimal solutions for a variety of geographical problems. This study will demonstrate that complex spatial search problems involving multiple-objectives and constraints can be conveniently tackled by using GAs and GIS.

2. Spatial search problems in GIS

2.1 *Heuristic search*

The search for optimal locations is a classical problem in the GIS domain. GIS has played a large role in the sitting of facilities for spatial decision making (Church 1999). For example, Openshaw and Steadman (1982) propose an optimal nuclear-bombing strategy based on population data. The optimal bombing problem involves defining an optimal set of targets that would cause maximum casualties. Casualty rules predict deaths from blast effects as a function of distance. The algorithm based on neighborhood search is given as follows (Openshaw and Openshaw 1997):

Step 1: Define casualty rules as a function of distance.

Step 2: Perform a 1 km grid mesh search for maximum casualty location. Find the first optimal target for bomb 1.
Step 3: Reduce population data by removing dead people.
Step 4: Repeat steps 2 and 3 for locating the optimal targets for *n* bombs separately.

The casualty rules are created using four concentric rings around ground zone with radii of 3.96, 6.42, 10.65 and 16.75 km. The population within each distance ring is computed and the blast death probabilities of 1.0, 0.52, 0.05 and 0.0 are applied respectively.

The above neighborhood search algorithm does not involve the combinations of various factors and parameters. Site selection can become very complicated when the effects of various factors are dealt with simultaneously instead of using each factor independently. The combination of factors usually cause the maximization problem to be high dimensional.

The exhaustive blind (brute-force) search of all possible combinations is a straightforward method for finding the best solution. It does not use information about the problem to direct the search. In many situations, the method is infeasible because the amount of computer time needed is extremely large (Openshaw and Openshaw 1997). Many complex search problems are a combinatorial explosion of the number of possible solutions that need to be investigated. Even a modern computer cannot complete the search within an acceptable time.

Heuristic search has been used in geography to find the approximate answer to difficult problems that cannot be given exact solutions. Usually there are a large, sometimes extremely large, number of possible solutions that may have to be examined. A heuristic method can be defined as a trick or rule of thumb that assists in solving a problem but no guarantee is given (Openshaw and Openshaw 1997). A spatial optimization problem may involve a very large search space. Heuristic search is a very important means of solving a broad range of spatial problems.

The algorithms for heuristic search are problem-related because a universal algorithm is not available. The simplest heuristic search is sequential (Openshaw and Openshaw 1997). For example, the target numbers can be sorted so that the total number of comparisons can be reduced. Another way for the heuristic search is to use the Monte Carlo optimization method. However, Monto Carlo optimization can get stuck in local suboptima because a move is only made when a better solution is found. Simulated annealing is a way to solve such problems (Aarts and Korst 1989). A good example of applying simulated annealing in spatial decision making is to solve high-dimensional and non-linear optimization problems for allocating land use efficiently (Aerts and Heuvelink 2002). The method is simple and efficient. Starting from an initial situation with 'energy level' $f(0)$, a small perturbation in the state of the system is brought about. This brings the system into a new state with energy level $f(1)$. If $f(1)$ is smaller than $f(0)$, then the state change is accepted. If $f(1)$ is greater than $f(0)$, then the change is accepted with a certain probability. However, jumping to a higher energy becomes less and less likely towards the end of the iteration procedure by gradually decreasing the freezing parameter. The whole procedure is repeated until the satisfied conditions have been reached.

Church (1999) identifies four general classes of location models: median, covering, capacitated, and competitive. A median model is to locate a fixed number of facilities under the condition that the average distance from any user to their closest facility is minimized. Covering models involve locating *n* facilities to cover all or most demand within some desired service distance, such as covering the maximum population. Capacitated models concern the limit that can be accomplished at each facility. Competition models allow a competitor to readjust to any location decisions other competitors have made. A recent development of location models is to incorporate GIS to provide detailed spatial information. However, the use of more detailed spatial information from GIS will cause the problem to be computationally burdensome (Church 1999).

Many applications require the consideration of these four issues to support spatial decision making. It is only a recent development for the integration of multiple-facility location models, like *p*-median and maximal covering (Church 1999). There is still a need to develop algorithms to solve capacitated facility location problems.

2.2 *Genetic algorithms for solving optimization problems*

A rapid developing method for solving optimization problems is based on the evolutionary approach. Genetic algorithms (GAs) have been applied to the solution of optimization problems in many disciplines (Goldberg 1989). One of the advantages of GAs is that specific programs are not required for seeking the optimal solution. This is very useful for dealing with many difficult spatial decision problems. The optimization procedure is based on the concept of natural selection. In an optimization problem, each parameter can be considered as a gene, which is simply represented by a finite sequence of 0's and 1's. A trial solution of a set of genes composes a chromosome. A number of different chromosomes form a population (individuals). Genes are interchangeable parts between two individuals.

The mechanism of creating a fitter generation is based on the adaptation of individuals. Fitness functions are used to evaluate the adaptation of each individual to the environment. Those individuals with higher fitness values are allowed to reproduce offspring (which can mutate after reproduction) with greater probabilities. As a result, they will breed more offspring. These most fit individuals are called elite individuals. Generating populations only from two parents may lose the best chromosome from the last population. Elitism is often used to avoid this problem. This means that at least one of the generation's best solutions is copied without any changes to a new population, so the best solution can survive to the succeeding generation.

This evolution process is repeated until some conditions are satisfied or the best solution is found. Generally, after dozens or even hundreds of generations, a good population eventually emerges. The individuals will solve the problem very well. In fact, the most fit (elite) individual will be an optimum or close to the optimum solution.

Genetic algorithms can be used to solve the spatial search problems of enormous possible combinations in an effective way. GAs are excellent for quickly finding an approximate global maximum or minimum value. They are bottom-up approaches which can produce more efficient search procedure compared with top-down heuristic methods. Moreover, the form of GAs is generally applicable to a variety of problems. The essence of GAs is to search the optimal solution using the operations of crossover and mutation.

Brookes (2001) proposes an interesting method to solve a spatial geometry problem. Genetic algorithm combined with a region-growing program is used to find the best configuration of patches subject to multiple criteria. A more recent approach is to use GAs to find a set of contiguous places that meet multiple optimization objectives (such as minimizing total cost and maximizing proximity to certain facilities (Xiao *et al*., 2002). It assumes that the locations of the facility are known before the optimization process. For example, planners may need to find a site for the construction of a residential building subject to the constraint that the site must be close to a shopping centre (the facility). The area of the contiguous site is required before running the model. This type of study mainly focused on the search for optimal shape parameters (for example, size, location, and orientation). Our study has a different focus in that GAs are used to find the best locations for a facility under various objectives, such as maximizing population coverage, minimizing the total transportation cost, and minimizing the proximity to roads using GIS; population and transportation data will be retrieved for the evolutionary approach from a GIS database.

3. The methodology

This study uses the evolutionary approach and geographical information for solving the problem of selecting multiple-sites. Site selection is a common procedure in GIS applications. The search for the best sites is usually required to minimize development impacts and raise efficiency in urban planning. The optimization problem needs to assess various candidate options. Each candidate option refers to a combination of n sites (targets) for accommodating n facilities (e.g. schools) in the space of $N \times N$ cells. The allocation of each site for the facility is associated with a certain amount of 'fitness' or 'benefit'. The total amount of 'fitness' can be assessed so that the best combination of sites can be determined.

The search space is extremely large when there are a large number of sites and available cells. This problem increases exponentially as more and more spatial details (e.g. population and transportation data) are provided. It is impractical to use conventional methods to solve the optimization problem which involves large amounts of spatial data. For example, the simple identification of the best sites for accommodating 20 facilities in the space of 100×100 cells or possible locations may involve a total number of $\frac{10000!}{20! \times (10000-20)!} = 4.03 \times 10^{61}$ combinations for brute-force search. Even a modern computer can hardly solve such a 'simple' question because the computation time is enormous. The search becomes further complicated when multiple-objectives and constraints are considered in the optimization process. The neighborhood search algorithm cannot be used to solve this type of optimization problem.

These problems can be tackled using the intelligent approach of GAs. Another unique feature for this proposed method is to exploit GIS for retrieving detailed spatial information. This study demonstrates that the optimal spatial allocation of n facilities can be automatically determined using the evolutionary approach. An optimal solution should generate the largest fitness (benefit) value. The problem is similar to the optimal nuclear bombing strategy which attempts to maximize the death toll for a given number of bombs (Openshaw and Steadman 1982). However, their strategy only considers the allocation of each bomb independently. The search algorithm cannot find the optimal solution because the combined effects of all the bombs are not addressed simultaneously. The proposed GA method has the advantage in dealing with the combined effects of all parameters together by properly devising chromosomes. The optimization procedures for allocating the n facilities (e.g. hospitals) based on the integration of GAs and GIS are shown in Figure 1.

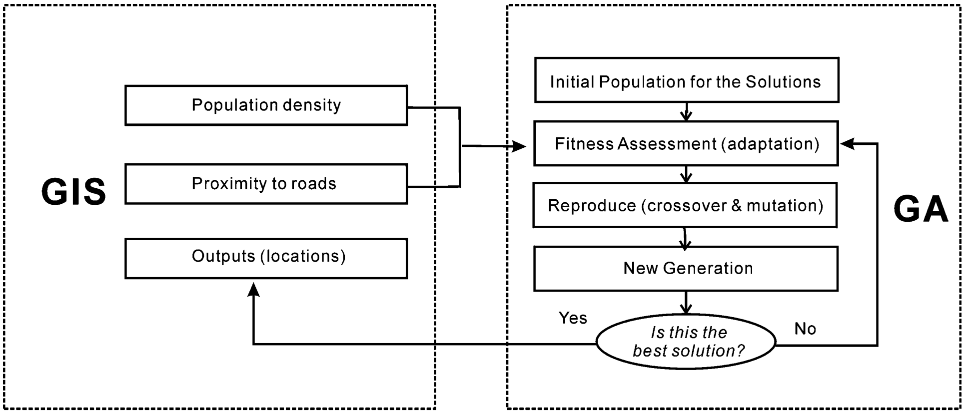

Figure 1. Integration of genetic algorithms and GIS for optimal location search

3.1 *Preparing spatial data from GIS*

GIS provide the detailed spatial data for the optimization process. Population data from the census department of Hong Kong are used to evaluate the fitness (benefit) of each candidate solution for allocating n facilities under various objectives. Population density is calculated using GIS functions. The final density layer will be converted into raster format to facilitate the calculation of fitness values. The proximity to roads is also calculated based on the transportation data using GIS.

3.2 *Encoding candidate solutions*

An important step for implementing GAs is to design chromosomes according to the problem domain. In this study, the site selection problem is to find out the optimal {x, y} coordinates for the n facilities within the spatial dimensions of $N \times N$ cells. A chromosome is devised to encode the combination location of n facilities (Figure 2). The chromosome has $2 \times n$ genes of which each represents one parameter of these coordinates. The chromosome (CM) is then expressed as follows:

$$CM = [x_1 y_1 x_2 y_2 x_3 y_3 \ldots x_n y_n] \tag{1}$$

where each pair of x_i and y_i represent the column and row numbers respectively for the location of a facility.

The program is to find out the optimal locations of the n facilities subject to the constraints from the GIS. In programming, these numbers are converted into a binary number of 0 and 1. It is very convenient to implement the crossover and mutation operations when the chromosome is expressed in the binary format (Figure 2).

3.3 *Defining fitness functions according to planning objectives*

The evolutionary process is mainly dependent on fitness functions. The functions should be used to assess the performance of each solution or individual (chromosome). It is obvious that fitness functions are crucial to the determination of the final results. There is no unique way to define fitness functions which should be related to problem domains. In many resource and environmental management

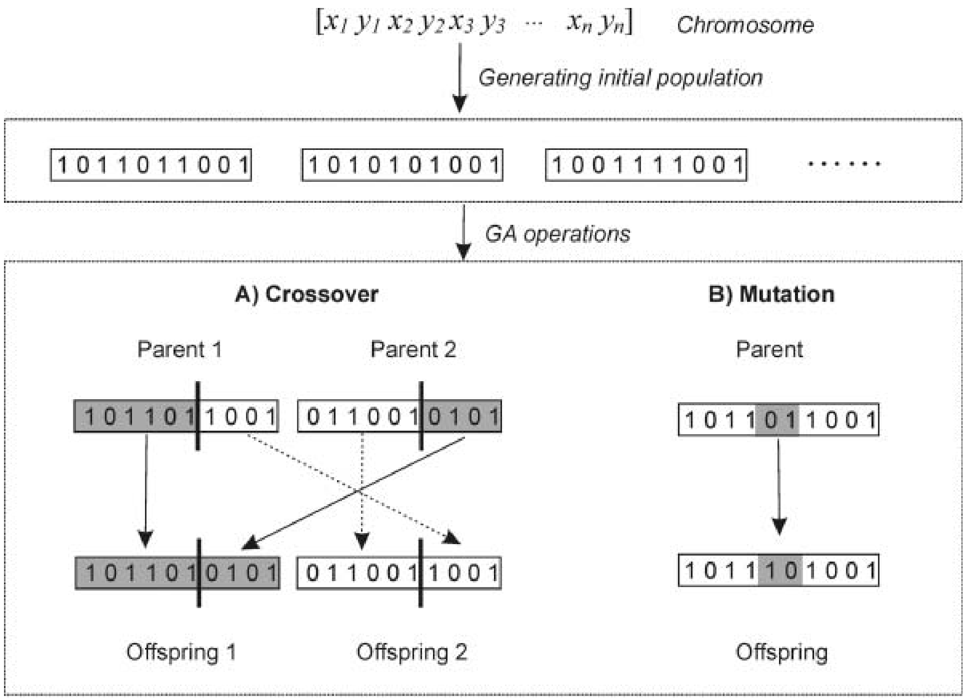

Figure 2. Operations of crossover and mutation for finding the best locations of facility

applications, fitness functions can be defined according to various planning objectives. In this way, planning alternatives can be conveniently generated.

This paper demonstrates that three planning objectives can be used to define fitness functions. The first two fitness functions correspond to single objective while the last fitness function corresponds to conflicting objectives. The use of conflicting objectives is rather common in planning practice. The procedure of incorporating other planning objectives is achieved by revising fitness functions appropriately.

Single Objective 1 – maximizing the population coverage. A simple fitness function is to calculate the total population served by a given n facilities. It is desirable to serve as much of the population as possible by a given n facilities. This is similar to the identification of the best targets for dropping n bombs with the objective of maximizing the death toll. The fitness can be calculated using the following equation:

$$F_1 = \sum_{i=1}^{n} \sum_{x=x_i-(l-1)/2}^{x_i+(l-1)/2} \sum_{y=y_i-(l-1)/2}^{y_i+(l-1)/2} P'_{den}(x,y) \times A_0 \times e^{-k\sqrt{(x-x_i)^2+(y-y_i)^2}} \tag{2}$$

where x_i and y_i are the coordinates (locations) for the ith site, n is the total sites of the facility to be allocated, l is the neighborhood window for summing up the total population served, $P'_{den}(x, y)$ is dynamic population density, A_0 is the area of each cell, $e^{-k\sqrt{(x-x_i)^2+(y-y_i)^2}}$ is the density decay function, and k is the coefficient of the decay function. The serving rate will decline away from a site of the facility. The dynamic population density is obtained by recalculating population density after a site has been allocated in each try. This can be done by just removing the served population from the current population.

Single Objective 2 – minimizing the total transportation costs. A number of additional fitness functions can also be defined based on various objectives. For example, the optimization problem can be based on the objective of minimizing the total transportation costs for all the population. The transportation costs are represented by summing the distance between the facility and each cell, which should be weighted by the population. This objective targets at minimizing energy consumption for transportation. Then the fitness function for this objective becomes:

$$F_2 = \frac{C_\lambda}{\sum_{x=1}^{N} \sum_{y=1}^{N} d_{\min}(x,y) \times P_{den}(x,y) \times A_0} \tag{3}$$

where $d_{\min}(x,y) = \min_i \left(\sqrt{(x-x_i)^2 + (y-y_i)^2} \right)$, C_λ is just a scaling constant, $P_{den}(x, y)$ is population density, and $N \times N$ is the size (the total number of rows and columns) of the study area.

Multiple Objectives – maximizing the population coverage, minimizing the total transportation costs and minimizing the proximity to roads. In most situations, multiple objectives are required to reflect users' preferences or knowledge in decision-making. These objectives can be complementary, but often conflicting. This means that a site can satisfy several objectives at the same time. For example, a site can be selected for development or conservation. The techniques to solve multiple-objective problems in GIS have been well developed and documented (Carver 1991, Jankowski 1995).

Multi-criteria evaluation (MCE) techniques can be used to deal with the problems of multi-criteria/objectives to support decision-making (Grabaum and Meyer 1998). Such approaches are often based on linear programming. The technique allows different objectives in a geographical region to be quantified and takes into account different weightings of scenarios. Site selection problems using GA can also be subject to multiple objectives by making multiple constraints for the evolutionary process. They can be combined into a single objective function using weights according to MCE techniques (Brookes 2001). Since the evolution process is controlled by the fitness function, the outcome should be the optimal solution to the multiple-objective problems.

This study demonstrates that the fitness function can be further devised to solve a multiple-objective problem in optimal location search. It includes three different objectives - maximizing the population coverage, minimizing the total transportation costs and minimizing the proximity to roads. The optimization problem involving combinations becomes quite complex when these three different objectives are used. Traditional methods have difficulty in finding the optimal solution. If the problem is to find a single site or multiple sites sequentially, multiple criterion evaluation (MCE) methods can be integrated with GIS to tackle the multiple-objective issue (Carver 1991). However, this method cannot solve the problems involving the combination of multiple-sites simultaneously, because the brute-force search for all the combinations is impossible. The incorporation of MCE within GA is required under this situation. The fitness function in the form of MCE is presented as follows:

$$
\begin{aligned}
F_3 &= \sum_i W_i X_i \\
&= W_1 X_{pop} + W_2 X_{trans} + W_3 X_{road} \\
&= W_1 + \sum_{i=1}^{n} \sum_{x=x_i-(l-1)/2}^{x_i+(l-1)/2} \sum_{y=y_i-(l-1)/2}^{y_i+(l-1)/2} P'_{den}(x,y) \times A_0 \times e^{-k\sqrt{(x-x_i)^2+(y-y_i)^2}} + \\
&\quad W_2 + \frac{C_\lambda}{\sum_{x=1}^{N}\sum_{y=1}^{N} d_{\min}(x,y) \times P_{den}(x,y) \times A_0} + W_3 \times \mathrm{e}^{-kD_{road}} \\
&= W_1 \times F_1 + W_2 \times F_2 + W_3 \times \mathrm{e}^{-kD_{road}}
\end{aligned}
\tag{4}
$$

where X_{pop}, X_{trans}, X_{road} are the variables (criteria) regarding the population coverage, total transportation costs and the proximity to roads, D_{road} is the distance to roads, and W_1, W_2 and W_3 are the weights of the above variables. These weights are usually decided by planners or experts to represent the priority or preference of each variable.

The individual scores of variables must be normalized before the calculation due is the use of different measurement scales. The normalization can allow the weights of each variable to be chosen properly. The normalization is usually done by scaling the variable value within the range of [0,1] using its minimum and maximum values. Since $F_1 \geqslant 0$ and $F_2 \geqslant 0$, the standardized fitness function can be represented as follows:

$$F'_3 = W_1 \times F_1/F_{1max} + W_2 \times F_2/F_{2max} + W_3 \times \mathrm{e}^{-kD_{road}} \tag{5}$$

Where F_{1max} and F_{2max} are the maximum values of F_1 and F_2, which cannot be found by normal procedures. However, they can be found by using the GA program separately from the solutions to the single objective problems using Equation (2) and (3). If there is no special preference for the weighting scheme, equal weights can be applied to the fitness function.

3.4 *Creating initial population for the candidate solutions*

An initial population is created by just using a random procedure. Each individual is a candidate solution (a string of coordinates). The population size should be determined for creating the breeding pool. However, there is no agreement on the size of the population for optimization procedures. If the size is set too small, there will not be enough individuals to find the best solution. If the size is too large, a much longer time is required for solving the problem. Usually, the population size between 20 and 200 individuals will yield good results. A larger size of population may be required when the problem is extremely complicated.

3.5 *Crossover and mutation operations*

There are two basic operations in the evolutionary approach - crossover and mutation (Figure 2). The GA program has used the binary string of 0, 1 for the easy operations of crossover and mutation. The conversion of the decimal figures into binary figures is automatically carried out by the VB program. The ‘crossover’ operator exchanges genes between two parents to form two offspring that inherit the traits of both parents. The cutting point for separating the genes is randomly decided. The 'mutation' operator alters one or more genes of a single parent. This can be done by randomly flipping bits from 0 to 1 or from 1 to 0. The effect of mutation is to prevent GA from stagnating at local optima or minima. It is expected that the new population (offspring) based on these two operations will be better than the old one because of the evolution process.

3.6 *Evaluating fitness and reproduction*

Each individual (a solution) corresponds to a fitness value. The evolutionary process is mainly based on the assessment of each individual using the fitness functions. The natural selection process is biased in favor of those individuals which have higher fitness values. The fitness functions determine which of the existing individuals are eliminated and replaced by the offspring of higher fitness from the reproductive process. The 'survival of the fittest' regime is crucial for reaching an optimum or near-optimum solution. The search process is intelligent because of the use of the evolutionary approach.

3.6 *Next generation*

The new generation will go through the same process as their parents until the best fit can be found in the population. In each generation, the best elite individual is identified among all the individuals. It is associated with the best fitness value. The best fitness value will increase with time as better offspring are produced by the evolutionary process. However, the increase of the best fitness value will be stabilized after many generations. The search procedure will stop when the improvement of the best fitness is insignificant. The rules for terminating the program are:

IF $F(t+1)-F(t)<\alpha$

THEN The search will be automatically terminated

where α is a small value.

The best elite individual at the latest search is then the final answer to the optimal allocations of the n sites of the facility. The coordinates for the optimal sites are given by the chromosome:

$$CM^0=\left[x_1^0y_1^0x_2^0y_2^0x_3^0y_3^0\ldots x_n^0y_n^0\right] \tag{6}$$

GA is plausible for solving four classical location problems - median, covering, capacitated, and competitive. The essential is to provide appropriate fitness functions. For example, the fitness functions in Equations (2) and (3) can guarantee that the maximum population coverage and the minimum total distance can be satisfied during the optimization process. Actually, some median models, which are based on Cooper's heuristic algorithm (1963 and 1967), cannot yield the optimal solution for achieving the average minimum distance because of using the median point. GAs can present much better results for finding optimal points. The capacitated objective can also be reflected by the density decay function in the equations. The approach can also allow the utility functions to be inserted in the location search by using them as the fitness functions. Moreover, the evolutionary approach is competition-oriented. This means that the all competitors (candidate locations) affect each other in the decision process. Moreover, the fitness function in Equation (5) is used to reflect conflicting or competitive objectives.

The optimization procedure is simple because complicated mathematical equations are not necessary. The only requirement is to modify the fitness function according to the problem domain. It has advantages over conventional mathematical methods. It is able to solve a lot of spatial optimization problems, such as searching for the optimal targets with the highest death toll for bombs, or allocating the optimal sites for feasible nuclear power plants. Moreover, geographical information can conveniently be used as inputs to the GA program.

4. Implementation and analysis results

4.1 *Study area and data*

The study area is located in the urban districts of Hong Kong. The proposed method is tested by solving an optimal problem which is to allocate n facilities (hospitals) across the region. This study will compare the effectiveness of using the proposed GA method, the neighborhood search algorithm, and simulated annealing for optimal location search.

The optimal allocation of the n facilities is based on the population data and transportation conditions. GIS was used to create the population layer from the 2001 population data (from the Hong Kong Census Statistics Department). The population data are available for district blocks. The population density was calculated for each polygon of the district blocks ARCGIS.

Hong Kong is one of the most densely populated areas in the world. Its land area is 1,101 km^2. The total population is 6.79 million and the average population density is 6,300 people/km^2 in 2002. The average population density in Kwun Tong of Kowloon is as high as 55,020 people/km^2. Figure 3A shows that the distribution of population is quite uneven in the region. The transportation conditions can be also prepared in the GIS by calculating the distances to major roads for each site (Figure 3B).

The population density and proximity to roads were prepared in a raster format that can be used for the optimization process. ARCGIS was used to produce the ASCII grid layers which have the cell size of 300 m^2 and the dimension of 150 cells $\times$ 150 cells (22,500 data points). The ASCII grids can be conveniently read by the program as input to the modeling process.

4.2 *Neighborhood search strategy*

A *neighborhood search* algorithm can be designed to obtain the approximate solution for allocating the n facilities (Openshaw and Openshaw 1997). Each site for the facility is decided separately. Although the maximum fitness (benefit) value can be obtained for each site independently, the total fitness value is not maximal because all the sites are not considered simultaneously. As a result, this neighborhood search method cannot guarantee that the solution can produce the maximum population coverage for a given number of facilities. The only advantages of this method are its simplicity and fast calculation speed, but its effectiveness for producing optimal results is in doubt for high-dimensional problems.

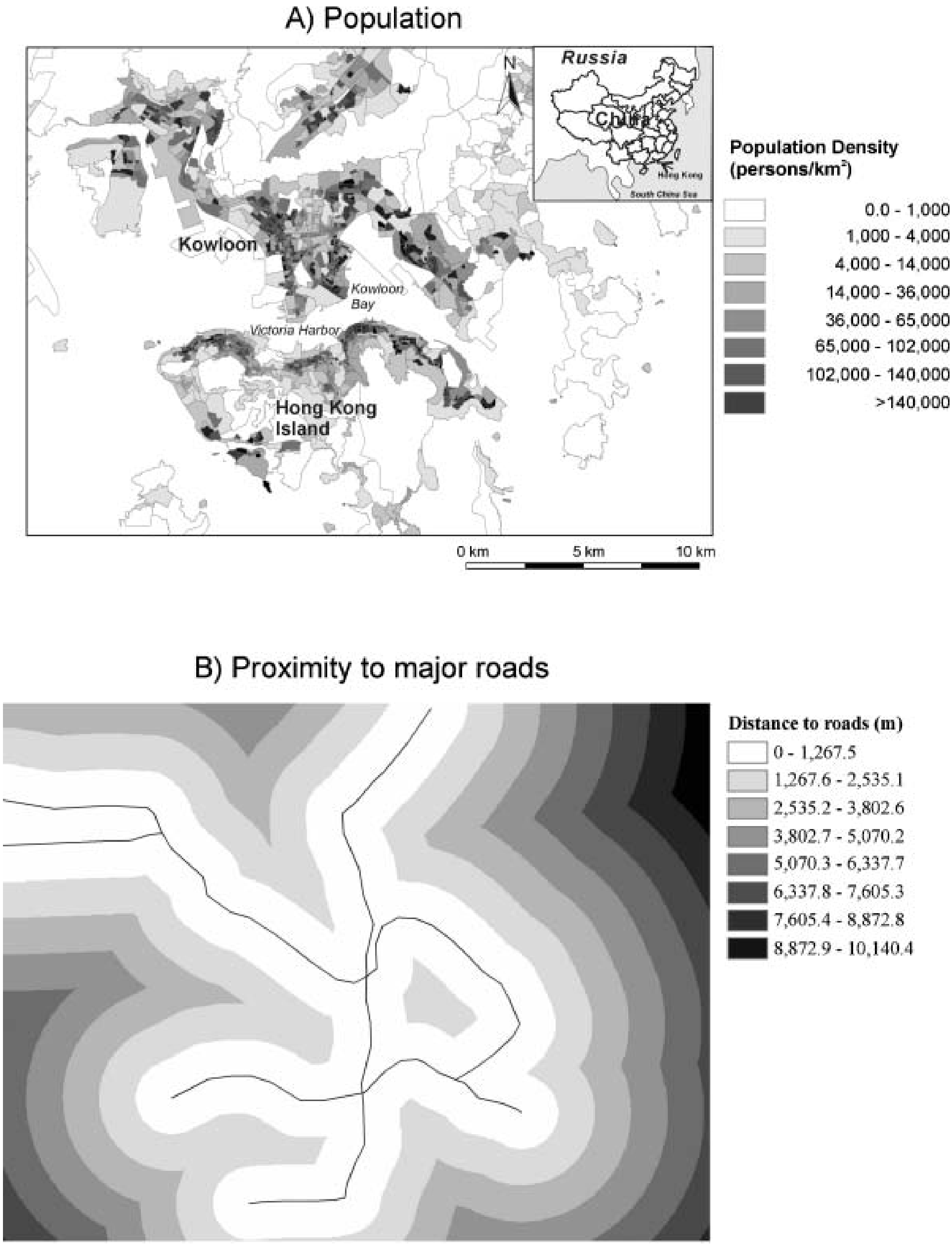

Figure 3. The population density and transportation conditions of the study area in Hong Kong

4.3 *Simulated annealing*

Another option is to use simulated annealing for solving the optimization problem of high dimensions (Aerts and Heuvelink 2002). First, initial locations for *n* facilities were randomly generated and the initial fitness value *F*(*0*) was calculated. A small perturbation was added to the coordinates of the initial locations, and a new fitness value *F(1)* determined. The acceptance of the change at each iteration was subject to the following rules by comparing the change of fitness values (Aerts and Heuvelink 2002):

If $F(t+1) > F(t)$ **Then**

The change is accepted

If $F(t+1) < F(t)$ and $\mathrm{Exp}(F(t+1) - F(t)/TC(t)) > Random[0,1]$ **Then**

The change is accepted

$TC(t)$ is the freezing parameter which should be gradually deceasing by using a multiple formula:

$$TC(t+1) = \delta\, TC(t) \tag{7}$$

where $0<\delta<1$. Typical values for the parameter are between 0.80 and 0.98 (Laarhoven 1987). The parameter was set to 0.9 in this experiment.

A parallel calculation was also used in the simulated annealing by using more than one solution at each iteration. A number of individuals were also randomly generated for the initial locations. The use of more than one solution can allow the algorithm to avoid the trapping in local optima.

4.4 *GA programming and parameters*

In this study, the GA program was developed using Visual Basic to solve the spatial optimization problem. A commercial genetic algorithm package, GeneHunter (Ward Systems Group 2004), was used to implement the evolutionary approach. The package provides the development toolkit that allows a user to utilize all functions in some common programming languages, such as Visual Basic. The basic functions of GAs can be called through the DLL (GALIB32.DLL) in a flexible way. This also allows GIS data to be conveniently incorporated in the optimization procedure.

In the GA programming, each of the individuals (chromosomes) in a population is a complete definition of a trial solution (e.g., a site selection plan). The chromosomes are encoded as a series of genes of which each defines a small part of the solution, such as a pair of {x, y} coordinates for the *i*th site in this study. The coordinates of *n* sites for allocating the facility are represented by a binary string (a chromosome). A fixed length bit string is used to represent the solution. For example, a bit binary string of the length, 8 bit $\times$ 2 $\times$ 10 sites $=$ 160 bits, is used to solve the site selection problem of allocating 10 sites of the facility. In this study, experiments were carried out for optimally allocating 2, 4, 6, 8, 10 and 12 facilities across the region respectively.

In the evolutionary approach, each individual (chromosome) is evaluated to determine its "fitness", which decides how likely the individual is to survive and breed into the next generation. New individuals are created according to the operations of crossover and mutation. The evaluation functions are inevitably domain dependent. In this study, three types of fitness functions were used to address both single-objective and multiple-objectives as described in Equation (2), (3) and (5).

The GA program requires the determination of a number of parameter values. Some are related to GAs themselves, and others are related to the fitness functions.

The first type of parameters includes population size, crossover rate and mutation rate. Usually, population sizes ranging from 20 to 200 can give a good result. Larger values of the population size may be used when the length of chromosomes is long. Experiments were carried out to determine the proper population sizes for the GA programming. Figure 4 compares the effects of various population sizes based on the linear weighted fitness function of maximizing population served, minimizing transportation costs, and minimizing proximity to roads as described in Equation (5). It is found that the best fitness values will be lower if the population size is either too small (population size=10) or too large (population size=300). The population size of 200 can yield the highest value of the best fitness. This means that the population size of 200 is more effective in finding the optimal solution because it can generate the highest value. The improvement of the best fitness value is stabilized after 400 generations for all the population sizes.

Usually, crossover is applied with a high probability while mutation is applied with a very low probability. Higher probability of mutation will reproduce random populations and will have problems in building up the evolutionary mechanism. In this study, the crossover rate is set to 0.98 and the mutation rate is set to 0.01. The scaling constant C_λ in Equation (3) is set to 10^{16} so that the values of the fitness functions can be scaled to a normal range – not too small and too large.

The parameters for the fitness functions include the neighborhood size (l) for summing the population, and the coefficient (k) for the distance decay function as described in Equation (2) and (4). In Equation (2), the fitness function has been formulated to count the total population served by n facilities. The population served can be calculated according to the distance decay function. The serving rate should be in an exponential decay function. Actually, a buffer window is used to calculate the total population served for convenience. In this study, the

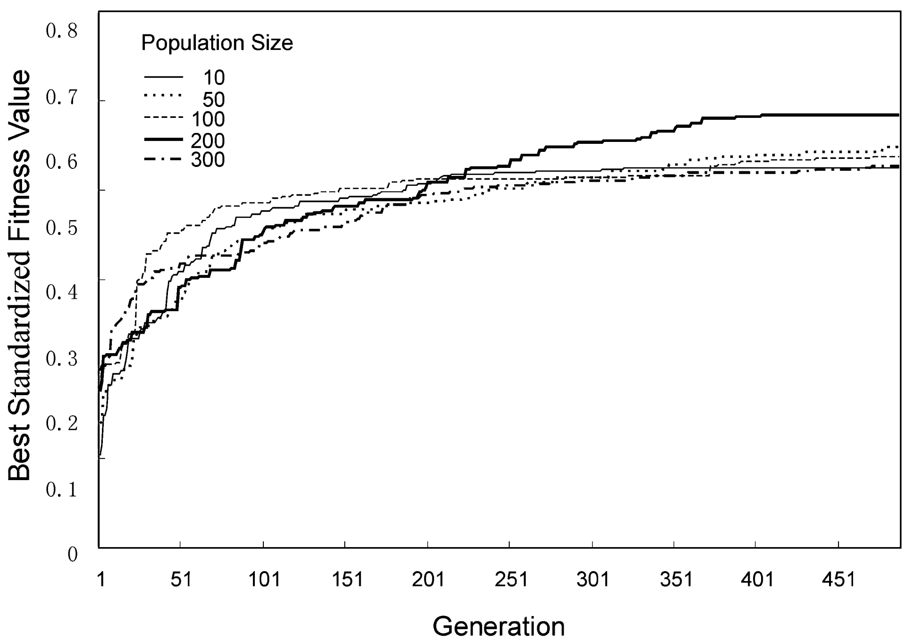

Figure 4. The relationships between the best standardized fitness value, generation and population size

neighborhood size (l) is set to 15 and the coefficient (k) is set to 0.05. Equal weights are applied to Equation (5) since no priority is given to any variable.

4.5 *Modelling results*

Comparison is made by applying the three methods - the neighborhood search algorithm, the simulated annealing, and the GA programming - to the study area. The comparison involves three objectives by using the fitness functions as described in Equation (2), (3) and (5). In the programming, a penalty function is also used for these three methods to avoid the selected sites falling into unreasonable area (e.g. sea).

The neighborhood search algorithm is unable to obtain the optimal results for the three fitness functions. It can only produce the approximate results based on the methodology. The approximate locations for n facilities can be calculated by using this strategy. The values for these three fitness functions can still be calculated when the locations of the facility are obtained.

A better method should be able to produce the highest value of the best fitness - the largest population served for a given number of facilities. Figure 5 just shows the allocation results of maximizing the population served by 10 facilities for these three methods. The neighborhood search method can find the approximate results very quickly because the algorithm is very simple. However, it cannot produce the optimal results because each selected site is independent of all others. The result of the neighborhood search algorithm can be regarded as the baseline for assessing the effectiveness of the proposed GA method and the simulated annealing. It is obvious that the proposed GA method has advantages because it is able to generate the largest value of fitness.

The proposed GA method has more advantages when the fitness function becomes more complex, such as the fitness functions as described in Equation (3) and (5). Figures 6 and 7 are the results of allocating 10 sites according to these two fitness functions respectively. They are to satisfy: 1) minimizing the total transportation costs; 2) maximizing the population served, minimizing the transportation costs, and minimizing proximity to roads at the same time. For the first option, the GA method produces the best fitness value (18.2% larger than the neighborhood search algorithm and 66.3% larger than the simulated annealing). For the second option, the GA method also has the best fitness value (30.7% larger than the neighborhood search algorithm, and 11.1% larger than the simulated annealing). Therefore, the proposed method can find the solution with much better fitness values than the other two methods. It is because the evolutionary approach is better at dealing with complex situations.

Figure 8 compares the improvement of the standardized fitness value with generations based on the multiple-objectives as described in Equation (5). Much better performance can be achieved by the GA method because it always has higher standardized fitness values.

Table 1 also clearly shows that improvement of the proposed GA method over the neighborhood search algorithm (baseline) for allocating various numbers of the facility. The proposed GA method has 24–42% improvement of standardized fitness values over the simple neighborhood search method.

The computation times of these three methods were also compared. Any optimization process should compare various possible combinations. For the present problem the brute force search for 12 sites is combinations. The calculation

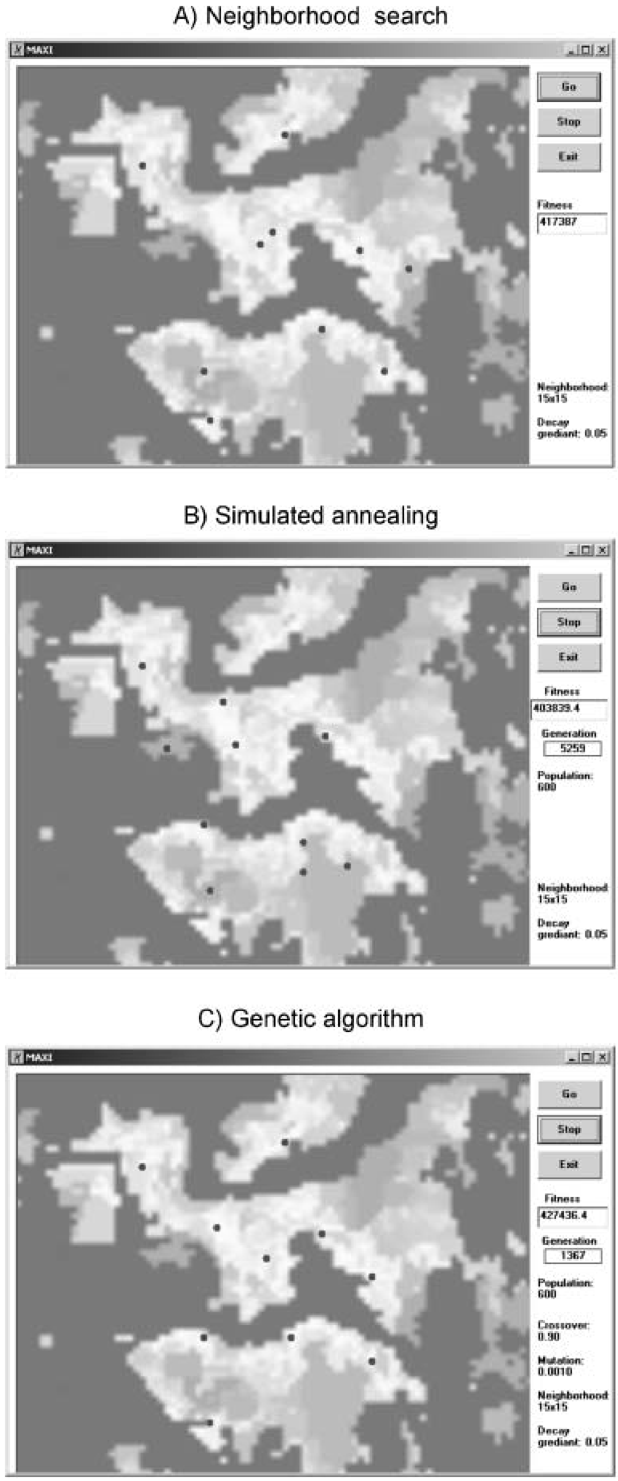

Figure 5. Optimal location search based on the fitness function of maximizing served population using neighborhood search, simulated annealing (SA), and generic algorithms (GA) (number of facility=10)

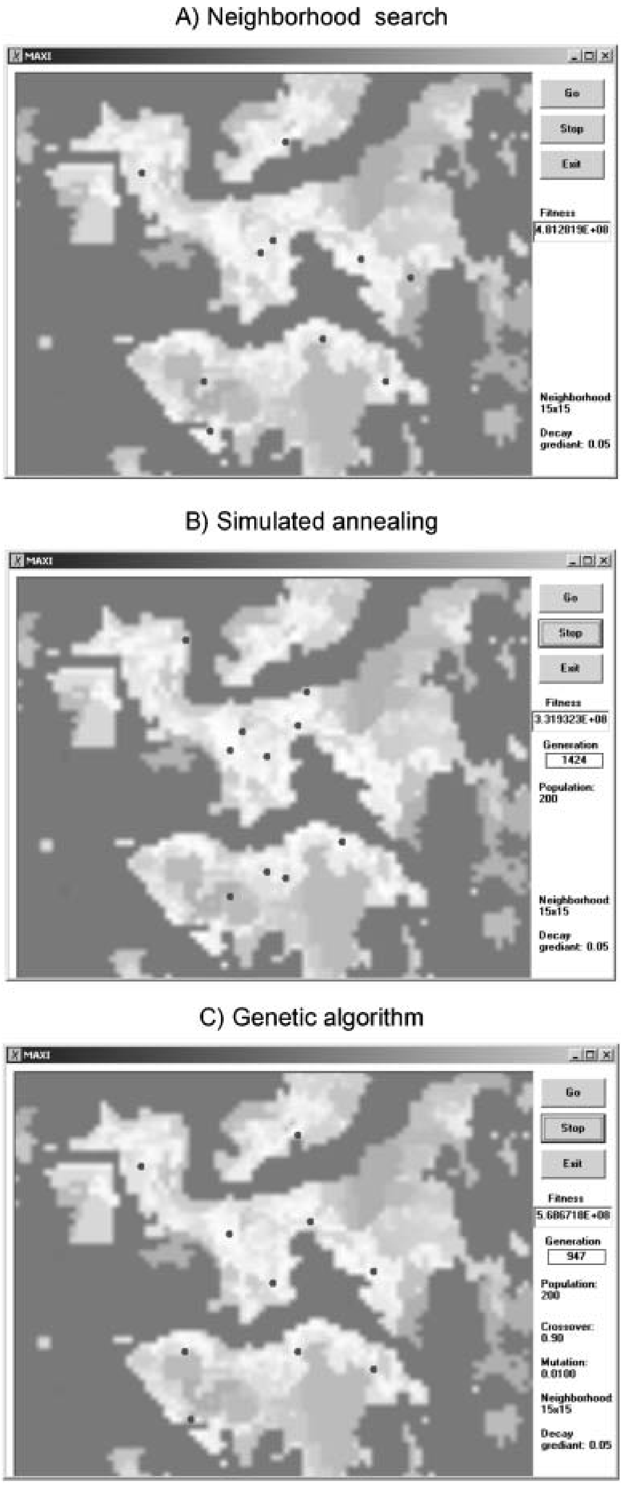

Figure 6. Optimal location search based on the fitness function of minimizing transportation costs using neighborhood search, simulated annealing (SA), and generic algorithms (GA) (number of facility=10)

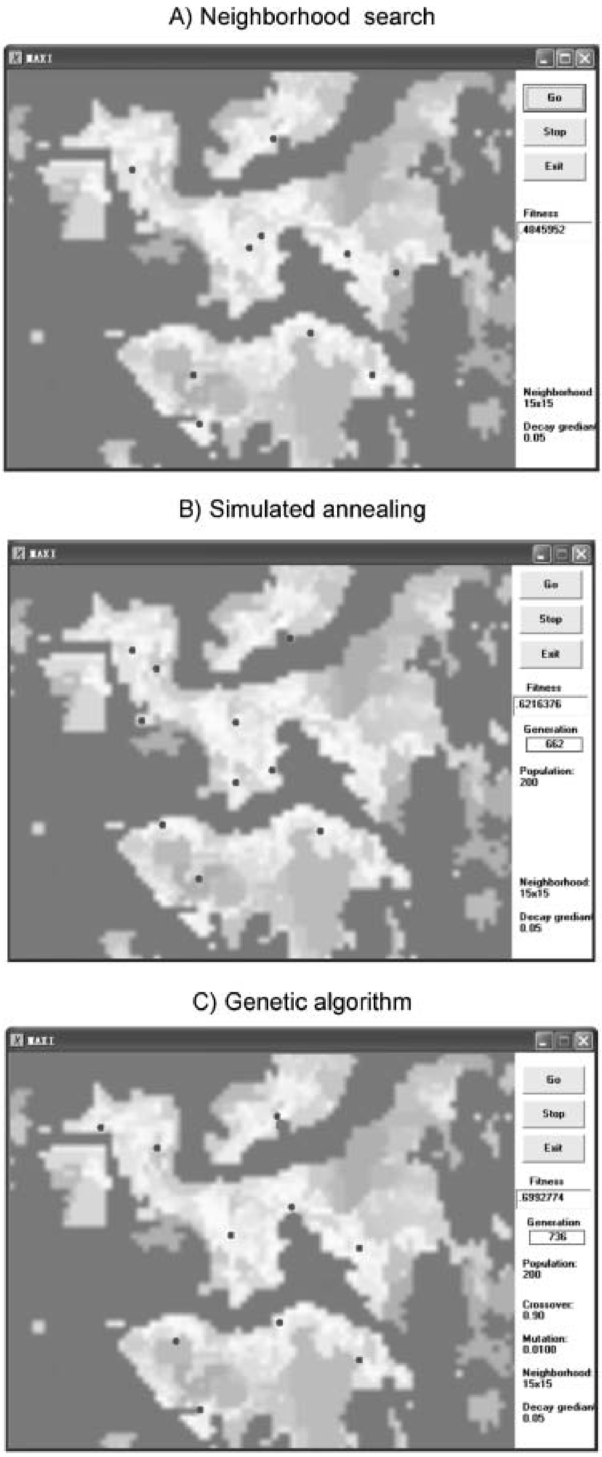

Figure 7. Optimal location search based on the standardized fitness function of maximizing served population, minimizing transportation costs and minimizing the proximity to roads using neighborhood search, simulated annealing (SA), and generic algorithms (GA) (number of facility=10)

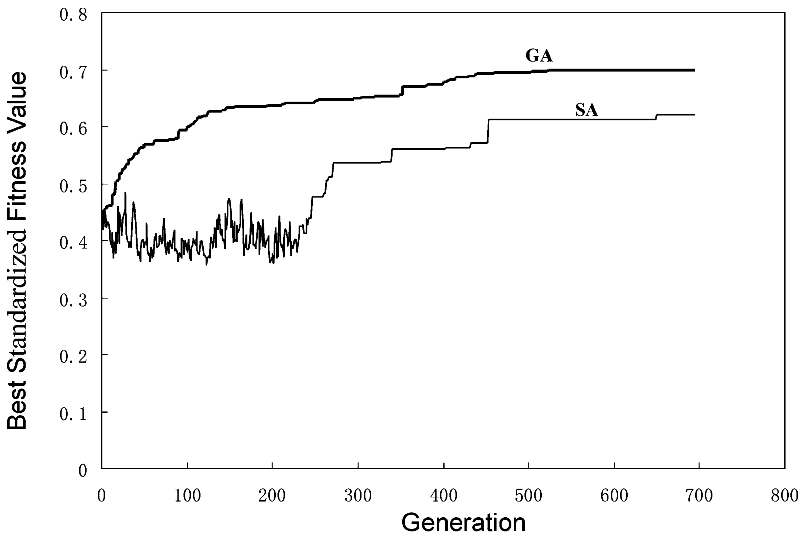

Figure 8. The comparison of the best standardized fitness values between simulated annealing (SA) and generic algorithms (GA)

Table 1. Comparison of the best fitness values between the GA method and the neighborhood search (baseline) based on the combined fitness function (F_3)

	The Best Standardized Fitness Values					
	2 sites	*4 sites*	*6 sites*	*8 sites*	*10 sites*	*12 sites*
Neighborhood search (Baseline)	0.466	0.423	0.533	0.496	0.485	0.454
GA method	0.765	0.734	0.758	0.652	0.699	0.711
Improvement of the GA method	39.1%	42.4%	29.7%	24.0%	30.7%	36.1%

of the fitness value each time takes about 0.15 seconds for a 1.5 GHz Pentium 4 processor. Therefore, the brute-force search will need 1.46×10^{39} hours to determine all combinations. The computation time is unacceptable even for a modern computer. Table 2 lists the computation time for various methods in allocating *n* facilities using the same computer. Although the neighborhood search is very quick,

Table 2. Computation time for various methods in the optimal location search process (hours)

	Number of Facility					
Methods	*2 sites*	*4 sites*	*6 sites*	*8 sites*	*10 sites*	*12 sites*
Neighborhood search	0.002	0.004	0.007	0.009	0.011	0.013
Brute-force (estimated)	$3.54e^{29}$	$2.95e^{30}$	$9.73e^{32}$	$1.77e^{33}$	$1.92e^{35}$	$1.46e^{39}$
Simulated Annealing	2.3	4.6	6.9	9.1	11.4	13.6
GA method	0.7	1.3	2.0	2.7	3.4	4.0

it is not an optimization procedure. The simulated annealing can explore various combinations, but it is not as efficient as the GA method. The computation time takes about 13.6 hours to allocate 12 sites of the facility whereas the GA method only takes about 29.4% of that time to finish the search.

5. Conclusion

This study demonstrates that genetic algorithms are capable of producing very satisfactory results for optimal location search under complex situations. This method has been tested by solving a spatial search problem which is to allocate a facility according to the population and transportation constraints derived from a GIS. The GA algorithm becomes very effective through the use of the mechanics of natural selection in biology. The proposed method can be used as a planning tool to solve location search problems under multiple-objectives. Potential applications may include the optimal sitting of public facilities, such as hospitals, schools, open space, and fire stations. The model is implemented in continuous space for better accuracy.

The proposed method has been tested in Hong Kong, a densely populated city. The study indicates that the proposed GA method can be conveniently integrated with GIS to retrieve spatial data. These spatial data are used to calculate the fitness values. The objective is to allocate n facilities (hospitals) across the region by maximizing a series of fitness (benefit) functions. This problem is impractical for the brute-force method because the combinations are enormous.

The proposed method is compared with neighborhood search and simulated annealing. This study demonstrates that neighborhood search method can only generate approximate results for selection of multiple sites. It can only process the search under simple assumptions and when multiple-sites and multiple-constraints are involved, the method cannot guarantee that the search results are optimal. Simulated annealing can deal with the optimization problems of high dimensions, but its performance is far below that of the proposed GA method. Much better performances can be obtained by using the proposed GA method under the same conditions. Furthermore, computation time of the GA method is only 29.4% of that of the simulated annealing.

MCE techniques can be incorporated in the GA program to deal with the issues of multiple-objectives. Optimal location search often requires the considerations of various planning objectives. These objectives can be combined into a single fitness function using a linear weighted equation in the GA program. The use of more objectives results in an increasingly complex search space in which traditional methods become less appropriate. The experiment indicates that the proposed method is well adapted to the solution of location search problems subject to multiple planning objectives. Much better performance can be achieved by applying the proposed method than either neighborhood search or simulated annealing. This method can be applied to solving a variety of facility siting problems, such as the optimal locations of schools, hospitals, power stations, and recreation centres.

Acknowledgement

This study is supported by the National Natural Science Foundation of China (NSFC) (projected No. 40471105), the '985' Project of GIS and Remote Sensing for Geosciences from the Ministry of Education of China, and the PhD development program from the Ministry of Education of China (Project No. 20040558023).

References

AARTS, E. and KORST, J., 1989, *Simulated Annealing and Boltzmann Machines: A Stochastic Approach to Combinatorial Optimization and Neural Computing (*Chichester, Sussex: John Wiley).

AERTS, C.J.H. and HEUVELINK, G.B.M., 2002, Using simulated annealing for resource allocation. *International Journal of Geographical Information Science*, **16**(6), pp. 571–587.

BROOKES, C.J., 2001, A genetic algorithm for designing optimal patch configurations in GIS. *International Journal of Geographical Information Science*, **15**(6), pp. 539–559.

CARVER, S.J., 1991, Integrating multi-criteria evaluation with geographical information systems. *International Journal of Geographical Information Systems*, **5**(3), pp. 321–339.

CHURCH, R.L., 1999, Location modeling and GIS. In Geographical information systems: Volume 1, edited by P.A. Longley, M.F. Goodchild, D.J. Maguire and D.W. Rhind (New York: John Wiley & Sons, Inc.), pp. 293–303.

COOPER, L., 1963, Location-allocation problems. *Operations Research*, **11**, pp. 331–343.

COOPER, L., 1967, Solutions of generalized location equilibrium problems. *Journal of Research Science*, **7**, pp. 1–18.

GOLDBERG, D.E., 1989, *Genetic Algorithms in Search, Optimisation and Machine Learning* (Reading, MA: Addison-Wesley).

GOODCHILD, M.F., 1979, Aggregation problem in location-allocation. *Geographical Analysis*, **11**(3), pp. 240–255.

GRABAUM, R. and MEYER, B.C., 1998, Multicriteria optimization of landscapes using GIS-based functional assessments. *Landscape and Urban Planning*, **43**, pp. 21–34.

HOLLAND, J., 1992, Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence (Cambridge, Mass: MIT Press).

JANKOWSKI, P., 1995, Integrating geographical information systems and multiple criteria decision-making methods. *International Journal of remote sensing*, **9**, pp. 251–273.

JIN, Y.Q. and WANG, Y., 2001, A genetic algorithm to simultaneously retrieve land surface roughness and soil wetness. *International Journal of remote sensing*, **22**(16), pp. 3093–3099.

LAARHOVEN VAN, P.J.M., 1987, *Theoretical and Computational Aspects of Simulated Annealing*, PhD Thesis, Erasmus University Rotterdam.

MITCHELL, M., 1996, *An Introduction to Genetic Algorithms* (Cambridge, Mass: MIT Press), 205p.

OPENSHAW, S. and OPENSHAW, C., 1997, *Artificial Intelligence in Geography* (Chichester: John Wiley & Sons).

OPENSHAW, S. and STEADMAN, P., 1982, On the geography of a worst case nuclear attack on the population of Britain. *Political Geography Quarterly*, **1**, pp. 263–278.

RECHENBERG, I., 1973, *Evolutionsstrategie: Optimierung technischer Systeme nach Prinzipien der biologischen Evolution* (Stuttgart: Fromman-Holzboog).

SIMHA, R., CAI, W.D. and SPITKOVSKY, V., 2001, Simulated N-body: new particle physics-based heuristics for a Euclidean location-allocation problem. *Journal of Heuristics*, **7**(1), pp. 23–36.

WARD SYSTEMS GROUP 2004, Gene Hunter. http://www.wardsystems.com/products.asp?p=genehunter, accessed 20 September 2004.

XIAO, N.C., BENNETT, D.A. and ARMSTRONG, M.P., 2002, Using evolutionary algorithms to generate alternatives for multiobjective site-search problems. *Environment and Planning A*, **34**, pp. 639–656.

ZHAN, H.G., LEE, Z.P., SHI, P., CHEN, C.Q. and CARDER, K.L., 2003, Retrieval of water optical properties for optically deep waters using genetic algorithms. *IEEE Transactions on Geoscience and Remote Sensing*, **41**(5), pp. 1123–1128.