We are IntechOpen, the world's leading publisher of Open Access books Built by scientists, for scientists

4,800
Open access books available

122,000
International authors and editors

135M
Downloads

Our authors are among the

154
Countries delivered to

TOP 1%
most cited scientists

12.2%
Contributors from top 500 universities

CLARIVATE ANALYTICS BOOK CITATION INDEX INDEXED

WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.
For more information visit www.intechopen.com

6

Knowledge-Informed Simulated Annealing for Spatial Allocation Problems

Jiunn-Der Duh
Department of Geography, Portland State University
Portland, Oregon, USA

1. Introduction

Generating prescribed patterns in spatial allocation is a difficult and complex optimization task. Many spatial allocation problems require the arrangement of resources in ways that their patterns promote some desirable landscape functions (e.g., Taylor et al.,2007; De Clercq et al., 2007; Milder et al., 2008). The complexity of the optimization task comes from the simultaneous effects of siting multiple spatial entities that usually require complex formulae to quantify (Tomlin, 1990; Brookes, 2001). Such spatial allocation problems are combinatorial in nature, and often require the use of global optimization algorithms such as simulated annealing or genetic algorithms (Revees, 1993) to find good solutions. Furthermore, spatial allocation problems often exhibit substantial complexity, especially when analyses must consider multiple, often conflicting, objectives (Malczewski, 1999). Despite successful examples of using global optimization algorithms in solving spatial allocation problems (Brookes, 2001; Aerts & Heuvelink, 2002; Xiao et al., 2002), however, an increase in the number of spatial entities involved in allocation deteriorates the performance of the trial-and-error mechanism of meta-heuristic algorithms.

Recent efforts to solve spatial optimization have been made by developing approaches that use auxiliary rules (i.e., heuristics; e.g., Church et al., 2003; Duh & Brown, 2005; Duh & Brown, 2007). Heuristic approaches, if used appropriately, can greatly improve the performance and utility of spatial optimization algorithms in spatial allocation and interactive spatial decision-making. This chapter describes the design, implementation, and evaluation of a knowledge-informed simulated annealing (KISA) algorithm that applies heuristics in single and multi-objective spatial allocation problems. The discussion at the end of the chapter addresses the potential applications and limitations of the approaches presented.

2. Spatial allocation problems

Spatial allocation is to arrange spatial entities in a two dimensional space so that the resulting arrangement exhibits certain preferred characteristics. The spatial entities involved in spatial allocation problems have been represented either as cells in a gridded coordinate system or as two dimensional geometric objects (i.e., polygons). These representations correspond to the raster and vector data models in Geographic Information Systems (GIS), which usually are used to model geographic phenomena as continuous fields and discrete

Source: Simulated Annealing, Book edited by: Cher Ming Tan, ISBN 978-953-7619-07-7, pp. 420, February 2008, I-Tech Education and Publishing, Vienna, Austria

objects respectively (Longley et al., 2005). The allocation problems discussed in this chapter was to assign cells in a raster data model with different land-cover types (e.g., trees and built-up areas). Contiguous cells of the same land-cover types form landscape *patches*. Mathematic formulae quantifying the patterns of landscape patches were used as objective functions of the spatial allocation problems. The approach introduced here can be implemented in vector-based data model with modifications.

There are various ways of quantifying patterns. Some examples are wavelet analysis (De Bonet & Viola, 1997), semivariance (Deutsch & Journel, 1992), Markov random field (Cross & Jain, 1983; German & German, 1984), and lacunarity (Dale, 2000; McIntyre & Wien, 2000). Recent development in landscape ecology provides a systematic way of understanding and quantifying landscape patterns in order to relate them to ecological or socioeconomic processes (e.g., Vos et al., 2001; Schmid-Holmes & Drickamer, 2001; McAlpine & Eyre, 2002; and Liu et al., 2003). The quantitative indices, also called landscape pattern metrics, provide ways of characterizing the composition or configuration, or both, of landscape patches on categorical maps (McGarigal & Marks, 1995). Because of the ecological implications of landscape pattern metrics, they could be used as the objective functions in spatial allocation problems that are intended to achieve ecological goals. I used a pattern metric that measures patch fragmentation as the pattern objective function of the optimization problem. The metric, PFF, developed by Riitters et al. (2000), is defined as the average proportion of cells among the eight neighboring cells of any cell of the same type (Equation 1).

$$\mathrm{PFF} = \frac{\sum_{i=1}^{N}\left[w_i \times \sum_{j=1}^{8} d_{ij} \right] / 8}{\sum_{i=1}^{N} w_i} \tag{1}$$

where d_{ij}, a neighborhood dummy variable, equals 1 when cell i and its neighbor j are of the same cover types, otherwise 0, w_i equals 1 when cell i is of the cover type currently measured, otherwise 0, and N is the number of pixels present in the landscape. PFF equals 0 when all pixels of a cover type, if there are any, are isolated, and 1 when the landscape is completely covered by a seamless cover of that cover type. Examples of landscapes with different PFF values are shown in Fig. 1.

Fig. 1. Example 18 by 18 landscape maps with 50% patch cells (dark color)

The single objective spatial allocation is formulated in a two dimensional space that is composed of N cells, of which K cells ($K < N$) are foreground whose pattern is to be measured and $N - K$ cells are background. The goal of the spatial allocation is to find the least fragmented landscape formed by a given number of patch cells. The problem is structured as follows:

Maximize PFF (2)

Subject to $$\sum_{i=1}^{N} w_i = K \quad (3)$$

In the multi-objective spatial allocation problem, in addition to the PFF pattern metric, a cost surfaces (C) was used to define a second objective function. The objective function was evaluated by summing the cell values on the cost surface if the location was occupied by foreground cells. The two-objective optimization problem is expressed as follows:

Maximize PFF (4)

Minimize $$\sum_{i=1}^{N} c_i \times w_i \quad (5)$$

Subject to $$\sum_{i=1}^{N} w_i = K \quad (6)$$

where c_i is the cost value at cell location i, w_i equals 1 when cell i is a foreground cell. When multiple objectives are specified in an optimization problem, finding one best solution that optimizes all objectives is often impossible, especially when the objectives to be achieved conflict with each other. Earlier generations of optimization algorithms dealt with multi-objective problems using a technique called scalarization, which collapses the multiple objectives to form a single objective (Sawaragi, Nakayama, et al., 1985; Eastman, Jin, et al., 1995). Such an approach involves the conversion of multiple objectives into commensurate criteria, which usually requires direct consultation with decision makers in finding the final solutions. This technique has several significant weaknesses: 1) it can only be applied to problems that are mathematically formulated; 2) it is inefficient when applied to large problems; and 3) it may fail to find important solutions (Miettinen, 1999). With the improvement of computers and algorithms, CPU-intensive approaches have been developed to find multiple compromise solutions (i.e., Pareto optimal solutions) that represent the trade-offs between conflicting objectives.

3. Knowledge-informed simulated annealing

3.1 Knowledge-informed algorithms

The term *knowledge-informed* optimization (Duh & Brown, 2005) is referred to as the algorithm that uses auxiliary knowledge (i.e., heuristics) of the nature and structure of spatial configuration to control the local search process in a global optimization algorithm. The main purpose of using auxiliary knowledge in a local-search algorithm is twofold: the auxiliary knowledge can reduce the search space, preventing unproductive search, or it can alter the structure of the solution space, making it easier to navigate to areas in the solution space where global optima are located. Empirical evidence indicates that optimization problems can thus be solved faster and easier (Pressey, Ferrier, et al., 1995; Sorensen & Church, 1996; Glover & Laguna, 1997). However, excessive use of inappropriate knowledge can generate significant errors in solving location problems and, very often, can result in sub-optimal solutions with different initial conditions (Church & Sorensen, 1996).

A simple but effective way to generate neighboring solutions in the local search process used in simulated annealing is by swapping cells randomly selected from the current best-solution. Knowledge-informed simulated annealing (KISA) uses some rules, instead of

complete random selections, to guide the generation of neighboring selections. The *compactness rule* (Duh & Brown, 2005) was used here to solve the spatial allocation problem whose goal was to find the least fragmented landscape. The KISA compactness rule preferentially move a randomly selected patch (i.e., foreground) cell to a location that promotes patch compactness, i.e., one with a high number of neighboring patch cells. PFF increases greatly when a patch cell is placed on a location where most neighbors are of the same cover type. Such an allocation not only increases the individual PFF for the patch cells being moved but also increases the individual PFF of the neighboring cells.

3.2 Multi-objective optimization

In recent decades, many techniques have been developed to address the needs of multi-objective decision-making. The most popular is the method of generating the efficient frontier, also known as the Pareto front. A Pareto front is formed by solutions whose performance on one objective cannot be improved without sacrificing performance on at least one other, a condition known as Pareto optimality (Pareto, 1971). A common way to determine Pareto optimality is using the concept of Pareto dominance. A Pareto optimal solution is referred to as a non-dominated solution.

In an optimization problem with D objectives, a solution x is said to dominate another x' (denoted $x \succ x'$) if and only if

$$f_i(x) \geq f_i(x') \ \ \forall i = 1,...,D \ \text{ and } \ f_i(x) > f_i(x') \ \text{ for some } i, \tag{7}$$

where $f_i(x)$ is the objective function value of objective i for a solution x. This formulation has assumed the problem is one of maximization, but the modifications necessary for a minimization problem are clear.

A set of solutions is said to be a non-dominated set (or Pareto set) if no member of the set is dominated by any other member. A non-dominated set is usually used as an approximation of the true Pareto front. The Automatic Accumulated Ranking Strategy (AARS) proposed by Goldberg (1989) provides a way to identify the non-dominated set in a set of solutions. AARS ensures that all the non-dominated solutions in the population are assigned rank 1 and removed from the population temporarily, then a new set of non-dominated solutions are assigned rank 2, and so forth. After all solutions have been assigned a rank, the solutions that have a Pareto ranking of 1 are non-dominated solutions. The pseudo-code of AARS algorithm is illustrated in Algorithm 1.

```
SUBROUTINE Set_Pareto_Ranking
BEGIN:
    Mark each solution in the solution set as not evaluated
    Set Current Ranking to 1
    EVALUATION:
    IF there are any not-evaluated solutions, THEN
        FOR EACH not-evaluated solution,  check If_solution_is_dominated
            IF it's not dominated, THEN
                Set its Pareto Ranking as the Current Ranking
                Mark the solution as evaluated
        After checking all solutions, increase Current Ranking by 1
    REPEAT EVALUATION
END
```

```
SUBROUTINE If_solution_is_dominated
BEGIN:
    FOR all other solutions in the solution set that are not evaluated or
    have a Pareto Ranking that equals the Current Ranking,
        Check if there exists at least one solution that has at least one objective function value
        that is larger (in the case of maximization) or smaller (in the case of minimization) than
        that of the target solution
    IF ''Yes'', THEN the solution is dominated, ELSE, the solution is not dominated
END
```

Algorithm 1. The pseudo-code of the Automatic Accumulated Ranking Strategy (Goldberg, 1989)

3.3 Multi-objective pareto simulated annealing

Multi-objective simulated annealing is conceptually identical to a single-objective simulated annealing algorithm. Czyzak & Jaszkiewicz (1998) modified simulated annealing algorithm for multi-objective optimization problems and developed Pareto simulated annealing (PSA). Instead of using just one candidate for the final solution, as done in the single-objective simulated annealing algorithm, PSA uses a set of interacting solutions, called the generating set *S*, at each iteration to propagate new solutions. The initial set of generating solutions is normally generated randomly. The subsequent sets of generating solutions are generated using a random swapping method based on the results at the prior stage. Any solution *y* generated that is not dominated by its preceding solution *x* in the generating set is checked for Pareto dominance among solutions in a non-dominated set *M*. The newly generated solution is added to the non-dominated set if it is non-dominated. All solutions originally in the non-dominated set that are dominated by the added solution are removed from the non-dominated set. PSA preserves some solutions based on a probability function *P*. The probability of preserving a new solution *y* in the generating set equals one when *y* dominates or is equal to the current solution *x*. Otherwise,

$$P(x, y, T, \Lambda^x) = \min\{1, \exp(\sum_{j=1}^{D} \lambda_j^x (f_j(x) - f_j(y))/T)\} \tag{8}$$

where $f_j(x) - f_j(y)$ is the change of the objective function values of objective *j* for solutions *x* and *y*, *D* is the number of objectives, *T* is the annealing temperature, and Λ^x is the weighting vector ($\Lambda^x = [\lambda_1^x, \lambda_2^x, ..., \lambda_D^x]$) used in the previous iteration for solution *x*. The weighting vector is used to assure dispersion of the generating solutions over the whole set of non-dominated solutions (i.e., the complete Pareto front). The higher the weight associated with a given objective, the lower the probability of accepting swappings that decrease the value on this objective and the greater is the probability of improving the value of this objective. For a given solution $x \in S$, the weights are changed in order to increase the probability of moving away from its closest neighbor in *S* denoted by *x'*. This is obtained by increasing the weights of the objectives with a factor of α ($\alpha > 1$ and is a constant close to 1) on which *x* is better than *x'* and decreasing the weights of the objective with a factor of $1/\alpha$ on which *x* is worse than *x'*. The general scheme of PSA is shown in Algorithm 2.

The PSA process is stopped when stop conditions are fulfilled. Several commonly used stop conditions include: 1) predetermined number of solutions (i.e., iterations) is generated and evaluated and 2) the accepting ratio of solutions falls below a threshold. When PSA stops, the non-dominated set *M* contains solutions that form the approximated Pareto front.

Select a starting set of generating solutions S
FOR each solution $x \in S$ DO
    Update the set M of potentially non-dominated solutions with x
Set current temperature T to initial temperature T_0

REPEAT
    For each $x \in S$ do
        Construct a feasible solution y
        IF y is not dominated by x THEN Update the set M with y
        Select the solution $x' \in S$ closest to x and non-dominated with respect to x
        IF there is no such x' or it's the 1st iteration with x THEN Set random weights such that

$$\forall j, \lambda_j^{x'} \geq 0 \text{ and } \sum_j \lambda_j^{x'} = 1$$

        Else
            For each objective f_j

$$\lambda_j^{x'} = \begin{cases} \alpha\lambda_j^x & f_j(x) \geq f_j(x') \\ \lambda_j^x / \alpha & f_j(x) < f_j(x') \end{cases} if$$

            Normalize the weights such that $\sum_j \lambda_j^{x'} = 1$

        Update x with y with acceptance probability $P(x,y,T, \Lambda^{x'})$
    If the conditions of changing the temperature are fulfilled then
    Decrease T according to cooling schedule $T(k)$
UNTIL the stop conditions are fulfilled

Algorithm 2. The pseudo-code of the Pareto simulated annealing algorithm (Czyzak & Jaszkiewicz, 1998)

3.4 Knowledge-informed pareto simulated annealing

There are two complementary knowledge-informed PSA strategies for improving the performance of PSA in solving multi-objective spatial allocation problems (Duh & Brown, 2007). First, similar to the single objective approach, auxiliary rules are used to preferentially generate subsequent solutions. Second, the Extended Initial Generating Set (EIGS) approach, which uses solutions optimized by single-objective simulated annealing as the initial solutions of PSA. This makes the initial generating set more diverse. The first strategy should result in an improvement in the effectiveness and efficiency of approximating the Pareto front. The second strategy, which extends the spread of the initial PSA generating set, is expected to encourage the diversity of Pareto solutions generated by PSA.

4. Performance evaluations

Multiple experiments were conducted to compare the performance of KISA against simulated annealing and Pareto simulated annealing in single and multi-objective spatial allocation problems. The allocation was carried out on a hypothetic 18 rows by 18 columns landscape (N = 324) with 50% of patch (i.e., foreground) cells (K = 162). K remained unchanged throughtout the simulation process. Quantitative performance indices were used for comparisons.

4.1 Single objective benchmark

In single objective experiments, ten random landscapes (K = 162) were used as initial maps. Each initial map was optimized ten times under each of four cooling schedules by simulated annealing (SA) and KISA with the compactness rule. The objective was to maximize PFF (see Section 2). Four cooling schedules used were: Boltzmann, Cauchy, Exponential, and Greedy. Their definitions are:

Boltzmann (logarithmic) schedule:

$$T(k) = \frac{T_0}{\ln k},\ k > 1. \tag{9}$$

Cauchy (modified logarithmic) schedule:

$$T(k) = \frac{T_0}{k},\ k > 0. \tag{10}$$

Quenching (exponential) schedule:

$$T(k) = T_0 \exp((c-1)k),\ c = 0.9966. \tag{11}$$

Greedy:

$$T(k) = 0 \tag{12}$$

In the equations above, T_0 is the initial temperature and k is an index of annealing time, which was defined as the number of processed iterations. The initial temperature was determined to allow about 80% of the deteriorated solutions be accepted initially (Laarhoven, 1987).

Two indices, Max Objective Function Value (MAXOFV) and Weighted Performance Index (WPI) (Duh & Brown, 2005), were respectively used to compare the effectiveness and efficiency between algorithms. MAXOFV is the best objecitve function value (OFV) ever achieved in each run. WPI is the average of weighted OFVs using a linearly decreasing weighting scheme, which gives more weight to the OFVs in the earlier stage of runs. These indices were calculated based on an arbitrary cutoff annealing time of 25000 iterations.

4.2 Multi-objective benchmark

For multi-objective experiments, the pattern metric, PFF, and two different cost surfaces, a uniform random and a conical surface (Fig. 2) were used as the objective functions. The random and conical cost surfaces exhibit low and high spatial autocorrelation of the distribution of cost, respectively. They were created as non-pattern objectives to contrast the PFF pattern objective. Costs are incurred when any location on the cost surface is occupied by a cell of the patch cover type. These objective functions formed two types of benchmark problems. The two types of problems represent the cases in which there are conflicting or concordant objectives. The first type (MOP1), maximizing PFF, which produces compact landscape, and minimizing the cost defined by the uniform random cost surface, which produces fragmented landscape, represents the case where the two optimization objectives are conflicting. The second type (MOP2), maximizing PFF and minimizing the cost defined by the conical cost surface, represents the case where the two objectives are concordant.

Fig. 2. Cost surfaces defined for multi-objective experiments: (A) uniform random and (B) conical (light color indicates higher costs if the location is occupied by patch cells).

Ten different initial landscape maps were used as the initial input to the multi-objective Pareto optimization. Four simulated annealing algorithms were tested. They are Pareto simulated annealing (PSA), knowledge-informed PSA (KIPSA), PSA with extended initial generating set (EIGS), and knowledge-informed PSA with extended initial generating set (KIPSA+EIGS). The size of generating set was ten. The ten generating solutions were created by randomly shuffling the initial map. For the Extended Initial Generating Set algorithm, only eight out of ten initial maps were randomly generated. Two additional initial maps were created to optimize each of the individual objectives specified in the problems and added to the generating set. I used the standard Boltzmann cooling schedule with an initial annealing temperature of 0.01. The values of both objectives were rescaled to the range 0 to 100 using their theoretical upper and lower bounds. Five repeated runs were conducted on each set of initial solutions, a total of 50 runs for each algorithm.

Two indices, Average Rank Value (RANK) and Average Spread Value (SPREAD), were used to measure the effectiveness of algorithms. RANK provides a relative comparison of the effectiveness of the algorithms for approximating the true Pareto front. It was calculated using the AARS method described earlier. The calculation first involved pooling the Pareto sets generated by different algorithms and assigning a Pareto ranking to every solution in the pool. The ranking values were then averaged for each algorithm to get the RANK index (i.e., average rank) of the corresponding algorithm. The closer the rank index value is to 1, the closer the corresponding Pareto set is to the true Pareto front. SPREAD is calculated based on the density evaluation scheme developed by Lu and Yen (2003). They calculated the density value by imposing a cell system, which is formed by partitioning the solution range of each objective into equally spaced cells, and counting the density of individual solutions within each cell. SPREAD is the quotient of the total number of solutions in a Pareto set and the average density value of the set. I randomly coupled individual runs of the four algorithms to create 50 combinations. Each of the 50 runs for a given algorithm was used exactly once. The two performance indices were calculated based on the 50 combinations of runs.

4.3 Random number generators

The simulated annealing algorithms use a random number generator (RNG) to control the stochastic process of local search. The inherent biases of RNGs could affect the outcomes of stochastic experiments (Van Neil & Laffan, 2003). Therefore, the validity of stochastic experiments is reliant on the RNG used. I tested the RNGs on the single-objective spatial allocation problem using, in addition to the rnd function in Microsoft Visual Basic (Microsoft, 1999), three other RNGs: Mother of All (MOA) (Marsaglia, 1994), Ranshi (Gutbrod, 1995), and Taus (L'Ecuyer, 1996, 1999) and found no systematic biases in performance. The results presented in this chapter were all derived from simulations based on the Visual Basic rnd RNG.

5. Results

A total of 800 runs were carried out in the single objective benchmark, 100 runs for each of the 8 combinations of the four cooling schedules and two algorithms. Both algorithms, simulated annealing (SA) and knowledge-informed simulated annealing (KISA), found near-optimal solutions (PFF > 0.94) and conspicuous sub-optimal solutions (PFF < 0.88) in some runs (Fig. 3). The average PFF at 25000 iterations was about 95% of the best PFF ever achieved at the maximal number of iterations (Table 1). The data confirm that most runs had converged at 25000 iterations and the use of 25000 iterations as the cutoff range for measuring MAXOFV and WPI was reasonable. KISA converged faster than SA in maximizing PFF (Fig. 4). When using the same cooling schedule, KISA performed better than SA, i.e., with higher MAXOFV values (Fig. 5a). The Boltzmann and the Exponential schedules were most effective but least efficient in generating the (near-)optimal solutions, whereas Cauchy and Greedy schedules, though more efficient in converging to optimal solutions, were not generating solutions as good as those generated using Boltzmann or Exponential schedules (Fig. 5b). The data suggest that using KISA with a Boltzmann or an Exponential cooling schedule is the most effective and efficient annealing setting for maximizing PFF.

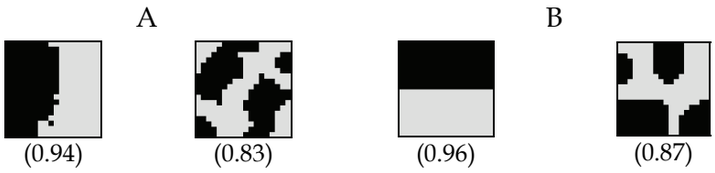


Fig. 3. Better and worse solutions and their PFF values (in parentheses) of maximizing PFF of patch class (dark color). These solutions are generated using (A) SA and (B) KISA.

Algorithm / *Schedule*	***SA***	***KISA***
Boltzmann	0.892 (0.945)	0.939 (0.958)
Cauchy	0.914 (0.914)	0.934 (0.956)
Exponential	0.896 (0.934)	0.944 (0.957)
Greedy	0.905 (0.917)	0.954 (0.958)

Table 1. The averaged maximal PFF reached in 25,000 iterations versus the maximal PFF ever reached in 150,000 iterations (in parentheses).

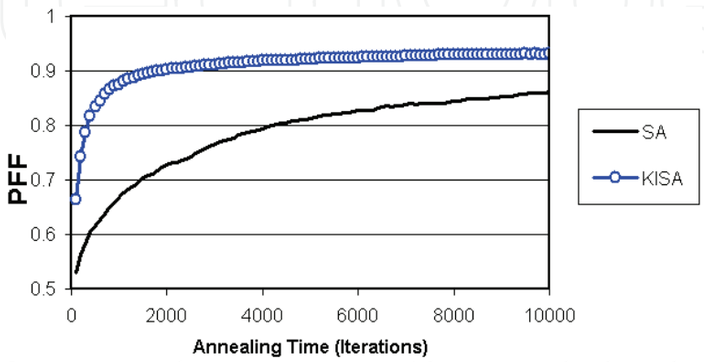


Fig. 4. Averaged OFV curves of maximizing PFF (showing only the solutions solved using the Boltzmann cooling schedule).

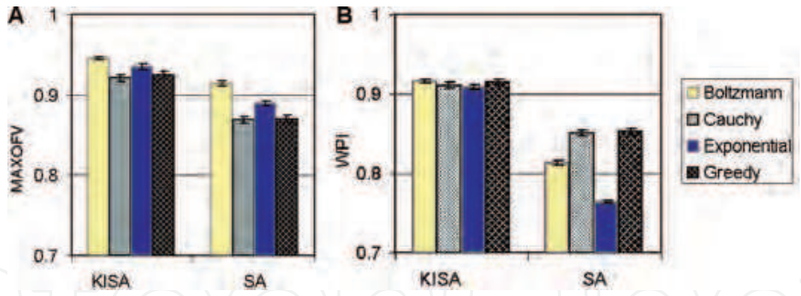


Fig. 5. Performance indices of maximizing PFF: (A) MAXOFV, (B) WPI. The error bars of ± 2 standard errors of the mean are included.

For multi-objective experiments, a total of 400 runs were carried out, 50 runs for each of the four algorithms in solving two types of multi-objective spatial allocation problems. The problems with conflicting objectives (i.e., MOP1) formed outstretched Pareto fronts (Fig. 6), while problems with concordant objectives (MOP2) formed compact Pareto fronts (Fig. 7). Approaches with extended initial generating set (i.e., EIGS and KIPSA+EIGS) have more outstretched Pareto fronts than those without EIGS (Fig. 6, 7). When look closely, in MOP1, KIPSA improved the approximations of the pattern objective but sacrificed by not exploiting the non-pattern objective (y-axis) as well as the PSA approach. Such sacrifices did not exist in MOP2 where the objectives were concordant.

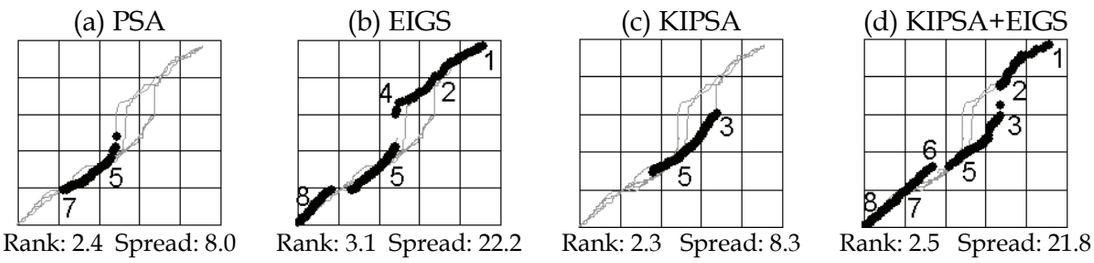


Fig. 6. Approximated Pareto fronts (gray lines) and Pareto solutions (black dots) derived using 4 PSA algorithms for MOP1. The Pareto set in each scatter plot represents the outcome of one run. The x-axis is the OFV for maximizing PFF and y-axis is the OFV for minimizing cost on the uniform random cost surface. Numbers indicate the locations of individual solutions shown in Fig. 8.

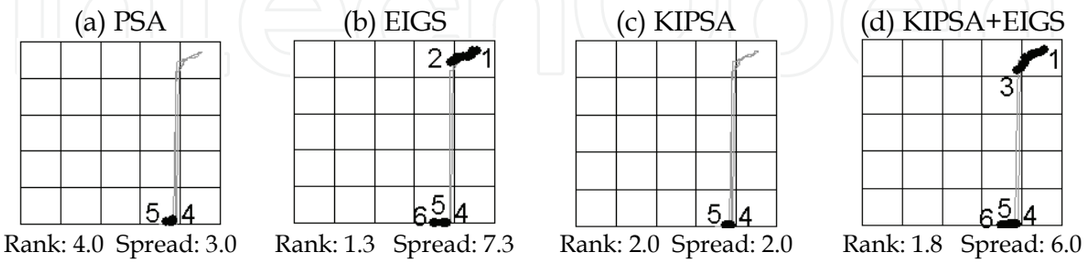


Fig. 7. Approximated Pareto fronts (gray lines) and Pareto solutions (black dots) derived using 4 PSA algorithms for MOP2. The Pareto fronts are superimposed as visual references. The Pareto set in each scatter plot represents the outcome of one run. The x-axis is the OFV for maximizing PFF and y-axis is the OFV for minimizing cost on the conical cost surface. Numbers indicate the locations of individual solutions shown in Fig. 9.

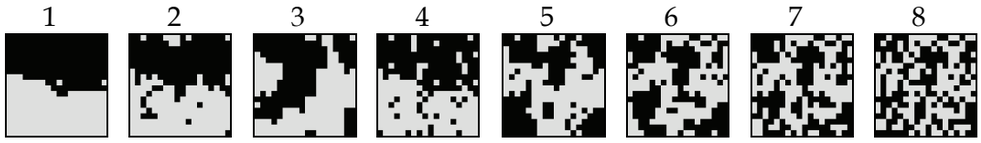


Fig. 8. Sample solutions of MOP1, each illustrates a particular combination of objective function values on a location indicated by a number shown in Fig. 6. These numbers do not connote that a solution was generated by a particular algorithm.

Fig. 9. Sample solutions of MOP2, each illustrates a particular combination of objective function values on a location indicated by a number shown in Fig. 7. These numbers do not connote that a solution was generated by a particular algorithm.

Sampled solutions for MOP1 (Fig. 8) illustrate the versatility of Pareto simulated annealing in solving multi-objective spatial allocation problems with conflicting objectives. Each map, 1 through 8, represents a (near-)optimal solution solved with different weightings of objectives in a single-objective optimization problem, with map 1 having a full weighting on maximizing PFF and with map 8 having a full weighting on minimizing the cost on the uniform random cost surface. The weighting on maximizing PFF diminishes from maps 1 to 8. Sample solutions for MOP2 (Fig. 9) indicate that, despite the capability of Pareto simulated annealing of generating diverse Pareto solutions, the diversity is intrinsic to the multi-objective optimization problems. Problems with concordant objectives have less diverse Pareto solutions.

The measures of performance indices reinforce and confirm the performance relationships alluded to above. Knowledge-informed algorithms (KIPSA and KIPSA+EIGS), in most cases, had significantly smaller average Pareto rankings (RANK) (Fig. 10a & 11a), indicating that KISA rule was more effective in generating Pareto solutions closer to the true Pareto front than PSA. However, knowledge-informed rules were not as effective in promoting diversity in Pareto solutions (Fig. 10b & 11b). The incorporation of EIGS greatly increased the spread of solutions (Fig. 10b & 11b).

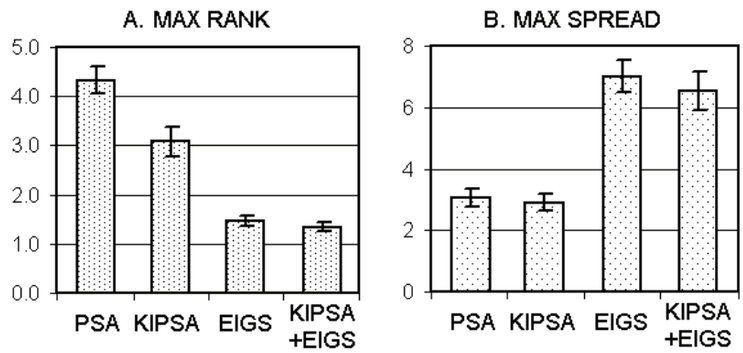


Fig. 10. Multi-objective performance indices for MOP1: (a) RANK, (b) SPREAD. The error bars of ±2 standard errors of the mean are included.

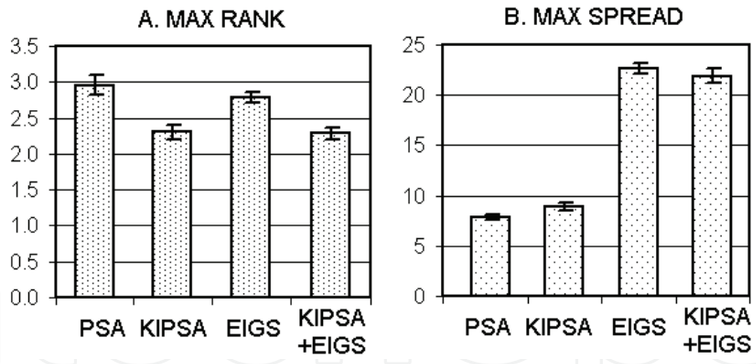


Fig. 11. Multi-objective performance indices for MOP2: (a) RANK, (b) SPREAD. The error bars of ±2 standard errors of the mean are included.

5. Discussion and conclusions

This chapter presents a knowledge-informed simulated annealing (KISA) approach to improving the performance of solving single and multi-objective spatial allocation problems. Simulated annealing is flexible and versatile in dealing with complex pattern objective functions, and empirical results indicate that KISA further improved its performance, making the approach of combining auxiliary information and simulated annealing desirable for similar applications. In addition to the compactness objective characterized by the PFF metric, there are other pattern objectives, such as connectivity. Corresponding KISA rules for these pattern objectives need to be designed, implemented, and evaluated.

The multi-objective benchmark shows that PSA algorithm is improved by various approaches, including using the KISA rule and extended initial generating sets (EIGS) strategy. The KISA rule improved the approximation of the Pareto front. EIGS greatly increased the diversity of Pareto solutions in problems with conflicting objectives. In these problems, efforts to approximate the Pareto front shifted toward the maximization of PFF when using the KISA rule, resulting in inferior approximations of the Pareto front toward the other objective, yet an overall improvement of the approximated Pareto front. One should use these strategies in multi-objective spatial optimization problems that emphasize pattern objectives.

The performance comparison in multi-objective benchmark did not measure the improvements in computation time. There is no predictable relation between the number of solutions evaluated and the CPU-time consumed for the algorithms used. This was because I did not set a maximal size of the Pareto set, so as the number of solutions in the Pareto set increases the required computation time for checking Pareto dominance also increases. It turned out that the PSA approaches that generated more diverse Pareto solutions used more CPU time and problems with conflicting objectives required more time to solve.

This research illustrates that knowledge-informed rules, which promote the formation of desirable pattern characteristics at an individual-cell level by acting through uncoordinated discrete steps, could eventually generate the desirable landscape patterns. Knowledge-informed simulated annealing should have the same performance improvement for other pattern metrics that were not tested in this research if the associated rules are tailored to capture the desirable pattern characteristics.

7. References

Aerts, J. C. J. H. & Heuvelink, G. B. M. (2002). Using simulated annealing for resource allocation. International Journal of Geographic Information Science, 16: 571-587.

Brookes, C.J. (2001). A genetic algorithm for designing optimal patch configurations in GIS. International Journal of Geographic Information Science, 15 (6): 539-559.

Church, R. L & Sorensen, P. (1996). Integrating normative location models into GIS, problems and prospects with the p-median model. In P. Longley & M. Batty (Eds.), Spatial Analysis: Modeling in a GIS Environment. Cambridge, UK: GeoInformation International.

Church, R. L., Gerrard, R. A., Gilpin, M., & Stine, P. (2003). Constructing cell-based habitat patches useful in conservation planning. Annals of the Association of American Geographers, 93 (4): 814-827.

Cross, G. R. and Jain, A. K. (1983). Markov random field texture models. IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI-5(1): 25-39.

Czyzak, P. & Jaszkiewicz, A. (1998). Pareto simulated annealing – A metaheuristic technique for multiple-objective combinatorial optimization. Journal of Multi-Criteria Decision Analysis, 7: 34-47.

Dale, M. R. T. (2000). Lacunarity analysis of spatial pattern: A comparison. Landscape Ecology, 15 (5): 467-478.

De Clercq, E. M., De Wulf, R., and Van Herzele, A. (2007). Relating spatial pattern of forest cover to accessibility. Landscape and Urban Planning, 80 (1-2): 14-22.

De Bonet, J. and Viola, P. (1997). A non-parametric multi-scale statistical model for natural images. In. Advance in Neural Information Processing, Vol 9. MIT Press.

Deutsch, C. V. and Journel, A. C. (1992). GSLIB: Geostatistical Software Library and User's Guide. Oxford University Press, Oxford.

Duh, J. D. & Brown, D. G. (2005). Generating prescribed patterns in landscape models. In D. J. Maguire, M. F. Goodchild, & M. Batty (Eds.), GIS, Spatial Analysis and Modeling (pp. 423-444), ESRI Press.

Duh, J. D. and Brown, D. G. (2007). Knowledge-Informed Pareto Simulated Annealing for Multi-Objective Spatial Allocation. Computers, Environment and Urban Systems, 31(3): 253-281.

Eastman, J. R., Jin, W. G., Kyem, P. A. K., & Toledano, J. (1995). Raster procedures for multi-criteria/multi-objective decisions. PE&RS, 61 (5): 539-547.

Geman, S. and Geman, D. (1984). Stochastic relaxation, Gibbs distribution and the Bayesian restoration in images. IEEE Trans. Patt. Anal. Mac. Int., 6(6): 721-741.

Glover, F and Laguna, M. (1997). Tabu search. Boston: Kluwer Academic Publishers.

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

Gutbrod, F. (1995). A fast random number generator for the Intel Paragon supercomputer. Computer Physics Communications, 87 (3): 291-306.

Laarhoven Van, P. J. M. (1987). Theoretical and Computational Aspects of Simulated Annealing. PhD Thesis, Erasmus University Rotterdam.

L'Ecuyer, P. (1996). Maximally Equidistributed Combined Tausworthe Generators. Mathematics of Computation, 65 (213): 203-213.

L'Ecuyer, P. (1999). Tables of Maximally-Equidistributed Combined LFSR Generators. Mathematics of Computation, 68 (225): 261-269.

Liu, Y. B., Nishiyama, S., and Kusaka, T. (2003). Examining landscape dynamics at a watershed scale using landsat TM imagery for detection of wintering hooded crane decline in Yashiro, Japan. Environmental Management, 31 (3): 365-376.

Longley, P. Goodchild, M. F., Maguire, D. and Rhind, D. (2005). Geographical Information Systems and Science. 2nd Edition. John Wiley and Sons, pp: 493-502.

Lu, H. M. & Yen, G. G. (2003). Rank-density-based multiobjective genetic algorithm and benchmark test function study. IEEE Transactions on Evolutionary Computation, 7 (4): 325-343.

Malczewski, J. (1999). GIS and Multicriteria Decision Analysis. John Wiley & Sons, Inc.

Marsaglia, G. (1994). The Mother of All Random Generators. Posted by Bob Wheeler to sci.stat.consult and sci.math.num-analysis on behalf of George Marsaglia on October 28, 1994. The code is available at ftp.taygeta.com.

McAlpine, C.A. and Eyre, T.J. (2002). Testing landscape metrics as indicators of habitat loss and fragmentation in continuous eucalypt forests (Queensland, Australia). Landscape Ecology, 17 (8): 711-728.

McGarigal, K. and Marks, B. J. (1995). FRAGSTATS: Spatial Pattern Analysis Program for Quantifying Landscape Structure. Gen. Tech. Report PNW-GTR-351, USDA Forest Service, Pacific Northwest Research Station, Portland, OR.

McIntyre, N. E. and Wiens, J. A. (2000). A novel use of the lacunarity index to discern landscape function. Landscape Ecology, 15: 313-321.

Microsoft (1999). MSDN Library Visual Studio 6.0 Release. Microsoft.

Milder, J. C., Lassoie, J. P., Bedford, B. L. (2008). Conserving biodiversity and ecosystem function through limited development: An empirical evaluation. Conservation Biology, 22 (1): 70-79.

Miettinen, K. M. (1999). Nonlinear Multiobjective Optimization. Kluwer Academic: Boston, MA.

Pareto, V. (1971). Manual of Political Economy (Augustus M. Kelley, New York) translated from French edition of 1927 by Ann S. Schwier, first published in 1896.

Pressey, R. L., Ferrier, S., Hutchinson, C. D., Sivertsen, D. P., & Manion, G. (1995). Planning for negotiation: using an interactive geographic information system to explore alternative protected area networks. In D.A. Saunders, J.L. Craig, & E.M. Mattiska (Eds.), Nature Conservation 4: The Role of Networks (pp. 23-33), Surrey Beatty, Sydney.

Revees, C.R. (1993). Modern Heuristic Techniques for Combinatorial Problems. John Wiley & Sons, Inc.

Riitters, K., Wickham, H., O'Neill, R., Jones, B., & Smith, E. (2000). Global-scale patterns of forest fragmentation. Conservation Ecology, 4 (2): 3.

Sawaragi, Y., Nakayama, H, & Tanino, T. (1985). Theory of Multiobjective Optimization. Academic Press: Orlando, FL.

Schmid-Holmes, S. and Drickamer, L. C. (2001). Impact of forest patch characteristics on small mammal communities: a multivariate approach. Biological Conservation, 99:293-305.

Sorensen, P. A. & Church, R. L. (1996). A comparison of strategies for data storage reduction in location-allocation problems. Geographical Systems, 3: 221-242.

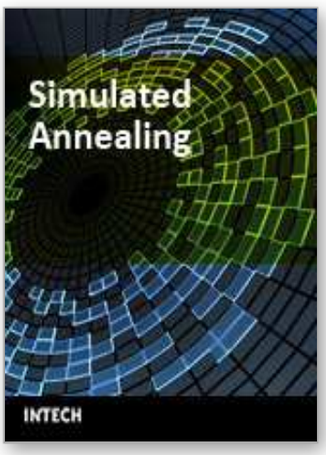
Taylor J. J. Brown, D. G. and Larsen, L. (2007). Preserving natural features: A GIS-based evaluation of a local open-space ordinance. Landscape and Urban Planning, 82: 1–16.

Tomlin, C.D. (1990). Geographic Information Systems and Cartographic Modelling. New Jersey: Prentice Hall.

Van Neil, K. & Laffan, S. W. (2003). Gambling with randomness: the use of pseudo-random number generators in GIS. International Journal of Geographic Information Science, 17 (1): 49-68.

Vos, C. C., Verboom, J., Opdam, P. F. M., and Ter Braak, C. J. F. (2001). Towards ecologically scaled landscape indices. American Naturist, 157(1): 24-51.

Xiao, N., Bennett, D.A., & Armstrong, M.P. (2002). Using evolutionary algorithms to generate alternatives for multiobjective site-search problems. Environment and Planning A, 34: 639-656.



Simulated Annealing

Edited by Cher Ming Tan

ISBN 978-953-7619-07-7
Hard cover, 420 pages
Publisher InTech
Published online 01, September, 2008
Published in print edition September, 2008

This book provides the readers with the knowledge of Simulated Annealing and its vast applications in the various branches of engineering. We encourage readers to explore the application of Simulated Annealing in their work for the task of optimization.

How to reference

In order to correctly reference this scholarly work, feel free to copy and paste the following:

Jiunn-Der Duh (2008). Knowledge-Informed Simulated Annealing for Spatial Allocation Problems, Simulated Annealing, Cher Ming Tan (Ed.), ISBN: 978-953-7619-07-7, InTech, Available from: http://www.intechopen.com/books/simulated_annealing/knowledge-informed_simulated_annealing_for_spatial_allocation_problems

INTECH
open science | open minds

InTech Europe
University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China
Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821

IntechOpen