19

A Comparison of Simulated Annealing, Elliptic and Genetic Algorithms for Finding Irregularly Shaped Spatial Clusters

Luiz Duczmal, André L. F. Cançado, Ricardo H. C. Takahashi and Lupércio F. Bessegato
Universidade Federal de Minas Gerais
Brazil

1. Introduction

Methods for the detection and evaluation of the statistical significance of spatial clusters are important geographic tools in epidemiology, disease surveillance and crime analysis. Their fundamental role in the elucidation of the etiology of diseases (Lawson, 1999; Heffernan et al., 2004; Andrade et al., 2004), the availability of reliable alarms for the detection of intentional and non-intentional infectious diseases outbreaks (Duczmal and Buckeridge, 2005, 2006a; Kulldorff et al., 2005, 2006) and the analysis of spatial patterns of criminal activities (Ceccato, 2005) are current topics of intense research. The spatial scan statistic (Kulldorff, 1997) and the program SatScan (Kulldorff, 1999) are now widely used by health services to detect disease clusters with circular geometric shape. Contrasting to the naïve statistic of the relative count of cases, the scan statistic is less prone to the random variations of cases in small populations. Although the circular scan approach sweeps completely the configuration space of circularly shaped clusters, in many situations we would like to recognize spatial clusters in a much more general geometric setting. Kulldorff et al. (2006) extended the SatScan approach to detect elliptic shaped clusters. It is important to note that for both circular and elliptic scans there is a need to impose size limits for the clusters; this requisite is even more demanding for the other irregularly shaped cluster detectors.

Other methods, also using the scan statistic, were proposed recently to detect connected clusters of irregular shape (Duczmal et al., 2004, 2006b, 2007, Iyengar, 2004, Tango & Takahashi, 2005, Assunção et al., 2006, Neill et al., 2005). Patil & Tallie (2004) used the relative incidence cases count for the objective function. Conley et al. (2005) proposed a genetic algorithm to explore a configuration space of multiple agglomerations of ellipses; Sahajpal et al. (2004) also used a genetic algorithm to find clusters shaped as intersections of circles of different sizes and centers.

Two kinds of maps could be employed. The point data set approach assigns one point in the map for each case and for each non-case individual. This approach is interested in finding, among all the allowed geometric shape candidates defined within a specific strategy, the one that encloses the highest ratio of cases vs. non-cases, thus defining the most likely cluster. The second approach assumes that a map is divided into M regions, with total population N and C total cases. Defining the zone z as any set of connected regions, the

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

objective is finding, among all the possible zones, which one maximizes a certain statistic, thus defining it as the most likely cluster. Although the first approach has higher precision of population distribution at small scales, the second approach is more appropriate when detailed addresses are not available. The genetic algorithms proposed by Conley et al. (2005) and Sahajpal et al. (2004), and also Iyengar (2004) used the point data set methodology.

The ideas discussed in this text derived from the previous work on the simulated annealing scan (Duczmal et al., 2004, 2006b), the elliptic scan (Kulldorff et al. 2006) and the genetic algorithm scan (Duczmal et al. 2007). The simulated annealing scan finds a sub-optimal solution trying to analyze only the most promising connected subsets of regions of the map, thus discarding most configurations that seem to have a low value for the scan likelihood ratio statistic. The initial explorations start from many and widely separated points in the configuration space, and concentrates the search more thoroughly around the configurations that show some increase in the scan statistic (the objective function). Thus we expect that the probability of overlooking a very high valued solution is small, and that this probability diminishes as the search goes on. Although the simulated annealing approach has high flexibility, the algorithm may be very computer intensive in certain instances, and the computational effort may not be predictable a priori for some maps. For example, the Belo Horizonte City homicide map analyzed in Duczmal et al. (2004) presented a very sharply delineated irregular cluster that was relatively easy to detect, with the relative risk inside the cluster much higher than the adjacent regions. This should be compared with the inconspicuous irregular breast cancer cluster in the US Northeast map studied in Duczmal et al. (2006b), which required more computer time to be detected, also using the simulated annealing approach. Although statistically significant, that last cluster was more difficult to detect due to the fact that the relative risk inside the cluster was just slightly above the remainder of the map. Besides, the intrinsic variance of the value of the scan likelihood ratio statistic for the sub-optimal solutions found at different runs of the program with the same input may be high, due to the high flexibility of the cluster instances that are admissible in this methodology. This flexibility leads to a very high dimension of the admissible cluster set to be searched, which in turn leads the simulated annealing algorithm to find sub-optimal solutions that can be quite different in different runs. These issues are addressed in this paper. We describe and evaluate a new approach for a novel genetic algorithm using a map divided into M regions, employing Kulldorff´s spatial scan statistic.

There is another important problem, common to all irregularly shaped cluster detectors: the scan statistic tries to find the most likely cluster over the collection of all connected zones, irrespectively of shape. Due to the unlimited geometric freedom of cluster shapes, this could lead to low power of cluster detection (Duczmal et al., 2006b). This happens because the best value of the objective function is likely to be associated with "tree shaped" clusters that merely link the highest likelihood ratio cells of the map, without contributing to the appearance of geographically meaningful solutions that delineate correctly the location of the true clusters. The first version of the simulated annealing method (Duczmal et al., 2004) controlled in part the amount of freedom of shape through a very simple device, limiting the maximum number of regions that should constitute the cluster. Without limiting appropriately the size of the cluster, there was an obvious tendency for the simulated annealing algorithm to produce much larger cluster solutions than the real ones. Tango & Takahashi (2005) pointed out this weakness, when comparing the simulated annealing scan with their flexible shape scan, which makes the complete enumeration of all sets within a

circle that includes the *k-1* nearest neighbors. Nevertheless, the size limit feature mentioned above was not explored in their numerical comparisons, thus impairing the comparative performance analysis of the algorithms. In Duczmal et al. (2006b) a significant improvement in shape control was developed, through the concept of geometric "non-compactness", which was used as a penalty function for the very irregularly shaped clusters, generalizing an idea that was used for the special case of ellipses (Kulldorff et al., 2006). Finally, the method proposed by Conley et al. (2005) employed a tactic to "clean-up" the best configuration found in order to simplify geometrically the cluster. It is not clear, though, how these simplifications impact the quality of the cluster shape, or how this could improve the precision of the geographic delineation of the cluster.
Our goal is to describe cluster detectors that incorporate the desirable features discussed above. They use the spatial scan statistic in a map divided into a finite number of regions, offering a strategy to control the irregularity of cluster shape. The algorithms provide a geometric representation of the cluster that makes easier for a practitioner to soundly interpret the geographic meaning for the cluster found, and attains good solutions with less intrinsic variance, with good power of detection, in less computer time. In section 2, we review Kulldorff's spatial scan statistic, the simulated annealing scan, the elliptic scan and the non-compactness penalty function. The genetic algorithm is discussed in section 3. The power evaluations and numerical tests are described in section 4. We present an application for breast cancer clusters in Brazil in section 5. We conclude with the final remarks in section 6.

2. Scan statistics and the non-compactness penalty function

Given a map divided into M regions, with total population N and C total cases, let the zone Z be any set of connected regions. Under the null hypothesis (there are no clusters in the map), the number of cases in each region follows a Poisson distribution. Define $L(Z)$ as the likelihood under the alternative hypothesis that there is a cluster in the zone Z , and L_0 the likelihood under the null-hypothesis. The zone Z with the maximum likelihood is defined as *the most likely cluster*. If μ_Z is the expected number of cases inside the zone Z under the null hypothesis, c_Z is the number of cases inside Z, $I(Z) = c_Z / \mu_Z$ is the relative incidence inside Z, $O(Z) = (C - c_Z)/(C - \mu_Z)$ is the relative incidence outside Z , it can be shown that

$$LR(Z) = L(Z) / L_0 = I(Z)^{c_Z} O(Z)^{C - c_Z}$$

when $I(Z) > 1$, and 1 otherwise. The zone that constitutes the most likely cluster maximizes the likelihood ratio $LR(Z)$ (Kulldorff, 1997). $LLR(Z) = \log(LR(Z))$ is used instead of $LR(Z)$.

2.1 The simulated annealing scan statistic

It is useful to treat the centroids of every cell in the map as vertices of a graph whose edges link cells with a common boundary. For the simulated annealing (SA) spatial scan statistic, the collection of connected irregularly shaped zones consists of all those zones for which the corresponding subgraphs are connected. This collection is very large, and it is impractical to calculate the likelihood for all of them. Instead we shall try to visit only the most promising zones, as follows (see Duczmal & Assunção (2004) for details). The zones z and w are neighbors when only one of the two sets w − z or z − w consists of a single cell. Starting

from some zone z(0), the algorithm chooses some neighbor z(1) among all the neighbors of z(0). In the next step, another neighbor z(2) is chosen among the neighbors of z(1), and so on. Thus, at each step we build a new zone adding or excluding one cell from the zone in the previous step. It is only required that there is a maximum size for the number of cells in each zone (usually half of the total number of cells). Instead of always choosing the highest LR neighbor at every step, the SA algorithm evaluates if there has been little or no LR improvement during the latest steps; in that case, the algorithm opts for choosing a random neighbor. This is done while trying to avoid getting stuck at LR local maxima.

We restart the search many times, each time using each individual cell of the map as the initial zone. Thus, the effect of this strategy is to keep the program openly exploring the most promising zones in the configuration space and abandoning the directions that seems uninteresting. The best solution found by the program is called a quasi-optimal solution and, for our purposes, it is a compromise due to computer time restraints for the identification of the geographical location of the clusters.

Duczmal, Kulldorff and Huang (2006) developed a geometric penalty for irregularly shaped clusters. Many algorithms frequently end up with a solution that is nothing more than the collection of the highest incidence cells in the map, linked together forming a "tree-shaped" cluster spread through the map; the associated subgraph resembles a tree, except possibly for some few additional edges. This kind of cluster does not add new information with regard to its special geographical significance in the map. One easy way to avoid that problem is simply to set a smaller upper bound to the maximum number of cells within a zone. This approach is only effective when cluster size is rather small (i.e., for detecting those clusters occupying roughly up to 10% of the cells of the map). For larger upper bounds in size, the increased geometric freedom favors the occurrence of very irregularly shaped tree-like clusters, thus impacting the power of detection. Another way to deal with this problem is to have some shape control for the zones that are being analyzed, penalizing the zones in the map that are highly irregularly shaped. For this purpose the geometric compactness of a zone is defined as the area of z divided by the circle with the perimeter of the convex hull of z. Compactness is dependent on the shape of the object, but not on its size. Compactness also penalizes a shape that has small area compared to the area of its convex hull. A user defined exponent a is attached to the penalty to control its strength; larger values of a increases the effect of the penalty, allowing the presence of more compact clusters. Similarly, lower a values allows more freedom of shape. The idea of using a penalty function for spatial cluster detection, based on the irregularity of its shape, was first used for ellipses in Kulldorff et al. (2006), although a different formula was employed.

We will penalize the zones in the map that are highly irregularly shaped. Given a planar geometric object z , define $A(z)$ as the area of z and $H(z)$ as the perimeter of the convex hull of z . Define the *compactness* of z as $\mathrm{K(z)} = 4\pi \mathrm{A(z)/H(z)^2}$. Compactness penalizes a shape that has small area compared to the area of its convex hull (Duczmal et al., 2006b). The strength of the compactness measure, employed here as a penalty factor, may be varied through a parameter $a \geq 0$, using the formula $K(z)^a$, instead of $K(z)$. The expression $LR(z)^{K(z)^a}$ is employed in this general setting as the corrected likelihood test function replacing $LR(z)$. The penalty function works just because the compactness correction penalizes very strongly those clusters which are even more irregularly shaped than the legitimate ones that we are looking for.

2.2 The elliptic scan statistic

Kulldorff et al. (2006) presented an elliptic version of the spatial scan statistic, generalizing the circular shape of the scanning window. It uses an elliptic scanning window of variable location, shape (eccentricity), angle and size, with and without an eccentricity penalty. An ellipse is defined by the x and y coordinates of its centroid, and its size, shape, and angle of the inclination of its longest axis. The shape is defined as the ratio of the length and width of the ellipse. For a given map, we define a finite collection of ellipses E as follows. For computational reasons, the shapes s in E are restricted to 1, 2, 4, 8 and 20. A finite set of angles is chosen such that we have an overlapping of about 70% for neighboring ellipses with the same shape, size and centroid. The ellipses' centroids are set identical to the cells' centroids in the map. We choose a finite number of ellipses whose sizes define uniquely all the possible zones z formed by the cells in that map whose centroids lie within some ellipse of the subset. The collection E is thus formed by grouping together all these subsets, for each cell's centroid, shape, and angle. We further define E(s) as the subset of E that includes all the shapes listed above in this section up to and including s. The choice of the collection E and its associated collection of zones is done beforehand and only once for a given map. The spatial scan statistic is thus applied to the collection of zones defined by E. The cluster likelihood was adjusted with a penalty function, the eccentricity penalty function

$$4s/(s+1)^2$$

so that the adjusted log likelihood is

$$LLR * [4s/(s+1)^2]^{\alpha}$$

and s is the cluster shape defined as the length of the longest axis divided by the length ofthe shortest axis of the ellipse. The tuning parameter α is similar to the parameter used in the simulated annealing scan.

3. The genetic algorithm approach

We approach the problem of finding the most likely cluster by a Genetic Algorithm specifically designed for dealing with this problem structure. Genetic Algorithms (GA' s) constitute a family of optimization algorithms that are devoted to find extreme points (minima or maxima) of functions of rather general classes.

3.1 The general structure of the genetic algorithm

A GA is defined as any algorithm that is essentially structured as:

- A set of N current candidate-solution points is maintained at each step of the algorithm (instead of a single current candidate-solution that is kept in most of optimization algorithms), and from the iteration to the next one the whole set is updated. This set is called the algorithm *population* (by analogy with a biological species population, which evolves according to natural selection laws), and each candidate-solution point in the population is called an *individual*.
- In an iteration, the algorithm applies the following *genetic operations* to the individuals in the population:
  - Some individuals (a subset of the population, randomly chosen) receive some random perturbations; this operation is called *mutation* (in analogy with the biological mutation);

- Some individuals (another random subset of the population) are randomly paired, and each pair of individuals (*parent individuals*) is combined, in such a way that a new set of individuals (*child individuals*, or *offspring*) is generated as a combination of the features of the initial ones. This is called *crossover* (in analogy with the biological crossover);
- After mutation and crossover, a new population is chosen, via a procedure that selects N individuals from ones that result from the mutation, from the crossover, and also from the former population. This procedure has some stochastic component, but necessarily attributes a greater chance of being chosen to the individuals with better objective function. This procedure is called the selection (by analogy with the natural selection of biological species), and results in the new population that will be subjected to the same operations, in the next iteration.

- Other operations can be applied, in addition to these basic genetic operations, including: the *elitism* operation (a deterministic choice of the best individuals in a population to be included in the next population); a *niche* operation (a decrement of the probability of an individual being chosen if it belongs to a region that is already covered by many individuals); several kinds of *local search*; and so forth.

Notice that the mutation introduces a kind of random walk motion to the individuals: an individual that were mutated iteration after iteration would follow a Markovian process. The crossover promotes a further exploitation of a region that is already being sampled by the two parent individuals. The selection introduces some direction to the search, eliminating the intermediate outcomes that don't present good features, keeping the ones that are promising. The search in new regions (mainly performed via mutation) and in regions already sampled (mainly performed via crossover) is guided by selection.

This rather general structure leads to optimization algorithms that are suitable for the optimization of a large class of functions. No assumption of differentiability, convexity, continuity, or unimodality, is needed. Also, the function can be defined in continuous spaces, or can be of combinatorial nature, or even of hybrid nature. The only implicit assumption is that the function should have some "global trend" that can be devised from samples taken from a region of the optimization variable space. If such a "global trend" exists, the GA is expected to catch it, leading to reasonable estimates of the function optima without need for an "exhaustive search".

There is a large number of different Genetic Algorithms already known and the number of possible ones is supposed to be very large, since each genetic operation can be structured in a large number of different ways, and the GA can be formed by any combination of operators. However, it is known that some GA's are much better than other ones, under the viewpoint of both reliability of solution and computational cost for finding it (Takahashi et al., 2003). In particular, for problems of combinatorial nature, it has been established that algorithms employing specific crossover and mutation operators can be much more efficient than general-purpose GA's (Carrano et al., 2006). This is due to the fact that a "blind" crossover or mutation that would be performed by a general-purpose operator would have a large probability of generating an unfeasible individual, since most of combinations of variables are usually unfeasible. Specific operators are tailored in order to preserve feasibility, giving rise only to feasible individuals, by incorporating the specific rules that define the valid combinations of variables in the specific problem under consideration. The GA that is presented here has been developed with specific operators that consider the structure of the cluster identification problem.

3.2 The offspring generation

We shall now discuss the genetic algorithm developed here for cluster detection and inference. The core of the algorithm is the routine that builds the offspring resultant from the crossing of two given parents. Each parent and each offspring is thus a set of connected regions in the map, or zone. We should associate a node to each region in the map. Two nodes are connected by an edge if the corresponding regions are neighbors in the map. In this manner, the whole map is associated to a non-directed graph, consisting of nodes connected by edges. Given the non-disjoint parents A and B, let $C = A \cap B$, and $D \subseteq C$ a randomly chosen maximal connected set. We shall now assign a *level*, that is, a natural number to each of the nodes of the parent A. All the nodes in D are marked as level zero. Define the *neighbors of the set U in the set V* as the nodes in V that are neighbors of some node belonging to U. Pick up randomly one neighbor x_1 of $A_0 = D$, $x_1 \in A - A_0$, and assign the level 1 to it. Then pick up randomly one neighbor x_2 of $A_1 = D \cup \{ x_1 \}$, $x_2 \in A - A_1$, and assign the level 2 to it. At the step n, pick up randomly one neighbor x_n of $A_{n-1} = D \cup \{x_{1,\ldots,} x_{n-1} \}$, $x_n \in A - A_{n-1}$, and assign the level n to it. In this fashion, choose the nodes, , $x_1,\ldots, x_m$ for all the m nodes of the set $A - D$ and assign levels to them. These m nodes, plus the virtual root node r , along with all the oriented edges (x_j, x_k) , where x_k was chosen as the neighbor of x_j in the step k ($j < k$) , and the oriented edges (r, x_k) , where x_k is a neighbor of D, forms an oriented tree T_A , with the following property:

Lemma 1: For each node $x_i \in A - D$ there is a path from the root node r to x_i, consisting only of nodes from the set $\{x_{1,\ldots,} x_{i-1} \}$.

Proof: Follow the oriented path contained in the tree T_A from r to x_i.

Note that the task of assigning levels to the nodes is not uniquely defined.

Repeat the construction above for the parent B and build the corresponding oriented tree T_B, but at this time using negative values -1,-2,-3,… for the levels, instead of 1,2,3,… (see the example in Figure 1). If $A - D$ and $B - D$ are non-disjoint, the nodes $y \in C - D$ are assigned with levels from both trees T_A and T_B (refer to Figure 1 again).

We now construct the *offspring of the parents A and B* as follows. Let $m_A \geq 2$ and $m_B \geq 1$ be respectively the number of elements of the sets $A - D$ and $B - D$, and suppose, without loss of generality, that $m_A \geq m_B$. The offspring is formed by the $m_B + (m_A - m_B - 1) = m_A - 1$ ordered sets of nodes corresponding to the sequences of levels (remembering that the level zero corresponds to the nodes of the set D):

$m_A - 1,\ldots,1,0,-1$

$m_A - 2,\ldots,1,0,-1,-2$

$\vdots$

$m_A - m_B,\ldots,1,0,-1,-2,\ldots,-m_B$

$m_A - m_B - 1,\ldots,1,0,-1,-2,\ldots,-m_B$

$\vdots$

$2,1,0,-1,-2,\ldots,-m_B$

$1,0,-1,-2,\ldots,-m_B$

If some sequence has two levels corresponding to the same node (it can happen only for the nodes in the set $C - D$), then count this node only once. Every set in the offspring has no more than $m_A + m_D$ nodes, where m_D is the number of nodes in D.

Lemma 2: All the sets in the offspring of the parents A and B are connected.
Proof: Apply lemma 1 to each node of each set in the offspring to check that there is a path from that node to the set D.
Figure 1A shows an example with two possible level assignments and their respective trees. The root node is formed by two regions. In the example of Figure 1B the set C is non-connected and consequently the node e has double level assignment. The successive construction of the ordered sets in the offspring requires a minimum of computational effort: from one set to the next, we need only to add and/or remove a region, simplifying the computation of the total population and cases for each set. Those totals are used to compute the spatial scan statistic. Besides, there is no need to check that each set is connected, because of lemma 2 (this checking alone accounted for 25% of the total computation time). Even more important is the fact that the offspring is evenly distributed along an imaginary "segment" across the configuration space, with the parents at the segment's tips, making easier for the program to stay next to a good solution, which could be investigated further by the next offspring generation.

3.3 The population evolution

The organization of the genetic algorithm is standard. We start with an initial population of M sets, or seeds, to be stored in the *current generation list*. Each seed is built through an aggregation process: starting from each map cell at a time, adjoin the neighbor cell that maximizes the likelihood ratio of the aggregate of cells adjoined so far, or exclude an existing one (provided that it does not disconnect the cluster), if the gain in likelihood ratio is greater; continue until a maximum number of cells is reached, or it is not possible to increase the likelihood of the current aggregate. In this fashion, the initial population consists of M (not necessarily distinct) zones, in such a way that each one of the M cells of the map becomes included in at least one zone.
We sort the current generation list in decreasing order by the LLR (modified as $\log(LR(z)^{K(z)^{\alpha}})$ in section 2), and pick up randomly pairs of parent candidates. If the conditions for offspring generation are fulfilled, the offspring is constructed and stored in an *offspring list*. This list is sorted in decreasing LLR order. The top 10% parents are maintained in the M-sized *new generation list*, and the remaining 90% posts of the list are filled with the top offspring population. At this step, *mutation* is introduced. We simply remove and add one random region at a small fraction of the new generation list (checking for connectedness). Numerical experiments show that the effect of mutation is relatively small (less than 0.1 in LLR gain for mutation rate up to 5%), and we adopt here 1% as the standard mutation rate. After that, the current generation list is updated with the LLRordered new generation list. The process is repeated for G generations.
We make at most tc_{MAX} tentative crossings in order to produce wsc_{MAX} well succeeded crossings (i.e., when $A \cap B \neq \phi$) at each generation. The graph of Figure 2 shows the results of numerical experiments. Each curve consists of the average of 5,000 runs of the algorithm, varying wsc_{MAX} and G such that $wsc_{TOTAL} = wsc_{MAX} * G$, the total number of well-succeeded crossings, remains equal to 4,000. Smaller wsc_{MAX} values cause more frequent sorting of the offspring, and also make the program to remove low LLR configurations faster. As a consequence, high LLR offspring is quickly produced in the first generations, at the expense

of the depletion of the potentially useful population with lower LLR configurations. That depletion impacts the increase of the LLR on the later generations, because it is more difficult now to find parents pairs that generate increasingly better offspring. Conversely, greater wsc_{MAX} values causes less frequent sorting of the offspring, lowering the LLR increase a bit in the first generations, but maintains a varied pool that produces interesting offspring, impacting less the LLR tax in the later generations. So, given the total number of well-succeeded crossings that we are willing to simulate, wsc_{TOTAL}, we need to specify the optimal values of wsc_{MAX} and G that produce the best average LLR increase. From the result of this experiment, we are tempted to adopt the following strategy: allow smaller values of wsc_{MAX} for the first generations and then increase wsc_{MAX} for the last generations. That will produce poor results, because once we remove the low LLR configurations early in the process, there will not be much room for improvement by increasing wsc_{MAX} later, when the pool is relatively depleted. Therefore, a fixed value of wsc_{MAX} is used.

4. Power and performance evaluation

In this section we build the alternative cluster model for the execution of the power evaluations. We use the same benchmark dataset with real data population for the 245 counties Northeastern US map in Figure 4, with 11 simulated irregularly shaped clusters, that has been used in Duczmal et al. (2006b). Clusters A-E are mildly irregularly shaped, in contrast to the very irregular clusters F-K. For each simulated data under these 11 artificial alternative hypotheses, 600 cases are distributed randomly according to a Poisson model using a single cluster; we set a relative risk equal to one for every cell outside the real cluster, and greater than one and identical in each cell within the cluster. The relative risks were defined such that if the exact location of the real cluster was known in advance, the power to detect it should be 0.999 (Kulldorff et al., 2003). Table 1 displays the power results for the elliptic, GA and SA scan statistics. For the GA and SA scans, for each upper limit of the detected cluster size, with (a=1) and without (a=0) noncompactness penalty correction, 100,000 runs were done under null hypothesis, plus 10,000 runs for each entry in the table, under the alternative hypothesis. The upper limit sizes allowed were 8, 12, 20 and 30 regions, indicated in brackets in Table 1. An equal number of simulations was done for the elliptic scan, for the E(1) (circular), E(2), E(4), E(8) and E(20) sets of ellipses, without using the eccentricity penalty correction (a=0).

The power values for the statistics analyzed here are very similar. For the SA and GA scans, the higher power values occur generally when the maximum size allowed matches the true size of the simulated cluster. For the elliptic scan, the maximum power was attained when the eccentricity of the ellipses matched better the elongation of the clusters.

The power performance was good, and approximately the same on both scan statistics for clusters A-E. The performance of the GA was somewhat better compared to the SA algorithm for the remaining clusters F-K, although the power was reduced on both algorithms for those highly irregular clusters. The GA performed generally slightly better for the highly irregular clusters I-K. For the clusters G (size 26) and H (size 29) the GA performance was better when the maximum size was set to 20 and 30, and worse when the maximum size was set to 8 and 12. For the clusters F and H, the GA performed generally slightly better using the full compactness correction (a=1) and worse otherwise (a=0).

cluster	size	E(1)	E(2)	E(4)	E(8)	E(20)	penalty	GA (SA) [8]	GA (SA) [12]	GA (SA) [20]	GA (SA) [30]
A	13	0.85	0.88	0.89	0.89	0.88	a=0	.84 (.87)	.84 (.86)	.79 (.79)	.68 (.66)
							a=1	.85 (.86)	.85 (.86)	.84 (.84)	.80 (.79)
B	16	0.79	0.83	0.84	0.82	0.80	a=0	.81 (.83)	.82 (.84)	.80 (.81)	.74 (.74)
							a=1	.81 (.78)	.84 (.84)	.86 (.86)	.84 (.83)
C	7	0.88	0.89	0.90	0.90	0.90	a=0	.87 (.87)	.86 (.84)	.82 (.77)	.72 (.65)
							a=1	.80 (.79)	.78 (.79)	.74 (.74)	.68 (.65)
D	15	0.86	0.89	0.90	0.90	0.88	a=0	.88 (.89)	.89 (.90)	.87 (.88)	.81 (.81)
							a=1	.86 (.85)	.89 (.89)	.90 (.90)	.87 (.87)
E	21	0.81	0.85	0.86	0.85	0.83	a=0	.83 (.82)	.86 (.85)	.87 (.87)	.84 (.84)
							a=1	.77 (.72)	.82 (.81)	.86 (.86)	.87 (.85)
F	23	0.70	0.72	0.75	0.75	0.73	a=0	.54 (.58)	.58 (.61)	.57 (.59)	.50 (.51)
							a=1	.45 (.44)	.46 (.45)	.48 (.46)	.44 (.44)
G	26	0.46	0.52	0.56	0.59	0.61	a=0	.58 (.61)	.62 (.63)	.66 (.62)	.68 (.59)
							a=1	.50 (.49)	.53 (.52)	.55 (.52)	.55 (.50)
H	29	0.66	0.69	0.71	0.72	0.71	a=0	.66 (.69)	.67 (.70)	.70 (.69)	.69 (.67)
							a=1	.64 (.62)	.66 (.67)	.67 (.67)	.64 (.64)
I	23	0.77	0.83	0.83	0.82	0.81	a=0	.66 (.65)	.71 (.67)	.74 (.69)	.71 (.67)
							a=1	.62 (.59)	.64 (.64)	.68 (.66)	.70 (.65)
J	55	0.68	0.71	0.72	0.72	0.70	a=0	.58 (.60)	.64 (.66)	.69 (.69)	.72 (.70)
							a=1	.56 (.54)	.62 (.63)	.68 (.67)	.68 (.67)
K	78	0.80	0.84	0.82	0.81	0.78	a=0	.53 (.51)	.61 (.60)	.69 (.68)	.75 (.72)
							a=1	.47 (.43)	.56 (.55)	.67 (.66)	.72 (.71)

Table 1: Power comparison between the elliptic scan (E), the genetic algorithm (GA), and the simulated annealing algorithm (SA), in parenthesis. For the last two methods, the noncompactness penalty correction parameter a was set to 1 (full correction) or 0 (no correction). The numbers in brackets indicate the maximum allowed size for the most likely cluster found.

The optimal power of the circular (E(1)) scan was above 0.83 for clusters A–E, I, and K, and below 0.75 for the remaining data with clusters F, G, H, and J. The performance was very poor on simulated data with cluster G, and the optimal power achieved was only 0.61, using the maximum shape parameter 20. Similar comments apply for clusters F, H, and J, with optimal power about 0.70 and maximum shape parameters 4 and 8. Better power was not achieved when we increased the maximum elliptic shape to 20 for these data. When clusters are shaped as twisted long strings, the elliptic scan tended to detect only straight pieces within them: this phenomenon was observed in clusters F, G, H, and J, resulting in diminished power. Otherwise, when a cluster fits well within some ellipse of the set, best power results were attained, as observed for the remaining clusters. The elliptic scan obtained somewhat better results for clusters A, C, F, I and K, which are easily matched by ellipses, and worse for the "non-elliptical" clusters E, G and J.
Numerical experiments show that the GA scan is approximately ten times faster, compared to the SA scan presented in Duczmal et al. (2004). For the GA, the typical running time for the cluster detection and the 999 Monte Carlo replications in the 72 regions São Paulo State map of section 5 and the 245 regions Northeast US were respectively 5 and 15 minutes with a Pentium 4 desktop PC. Using exactly the same input for 5,000 runs for both the GA and SA scans, calibrated to achieve the same LLR average solution values in the Northeast US map under null hypothesis, we have verified that the GA sub-optimal solutions have about five times less LLR variance compared to the SA scan approach.

5. An application for breast cancer clusters

The genetic algorithm is applied for the study of clusters of high incidence of breast cancer in São Paulo State, Brazil. The population at risk is 8,822,617, formed by the female

population over 30-years old, adjusted for age applying indirect standardization with 4 distinct 10 years age groups: 30-39, 40-49, 50-59, and 60+. In the 4 years period 2000-2003, a total of 14,831 cases were observed. The São Paulo State map was divided into 72 regions. The breast cancer data was obtained from Brazil´s Ministry of Health DATASUS homepage (www.datasus.gov.br) and de Souza (2005). Figure 3A shows the relative incidence of cases for each region, where the darker shades indicate higher incidence of cases. The other three maps (Figures 3B-D) show respectively the clusters that were found using values 1.0, 0.5 and 0.0 for the parameter a, which controls the degree of geometric shape penalization. Using 999 Monte Carlo replications of the null hypothesis, it was verified that all the clusters are statistically significant (p-values 0.001). The maximum size allowed was 18 regions for all the clusters. Notice that when a = 1.0 the cluster is approximately round, but with a hole, corresponding to a relatively low count region that was automatically deleted. As the value of the parameter a decreases we observe the appearance of more irregularly shaped clusters. As more irregularly shaped cluster candidates are allowed, due to the lower values of the parameter a, the LLR values for the most likely cluster increase, as can be seen in Table 2. The case incidence is about the same in all the clusters, by Table 2. It is a matter of the practitioner's experience to decide which of those clusters is the most appropriate in order to delineate the "true" cluster. The cluster in Figure 3B should be compared with the primary circular cluster that was found by SatScan (the rightmost circle in Figure 3D). It is also interesting to compare the cluster in Figure 3D with the primary and secondary circular clusters that were found by the circular SatScan algorithm (see the circles in Figure 3D).

Figure	A	Size	Cases	Population	Incidence	LLR	p-value
3B	1.0	16	3,324	394,294	0.00843	298.9	0.001
3C	0.5	16	3,078	361,373	0.00852	343.8	0.001
3D	0.0	18	2,924	346,024	0.00845	449.6	0.001

Table 2. The three clusters of Figure 3B-D.

6. Conclusions

We described and evaluated a novel elitist genetic algorithm for the detection of spatial clusters, which uses the spatial scan statistic in maps divided into finite numbers of regions. The offspring generation is very inexpensive. Children zones are automatically connected, accounting for the higher speed of the genetic algorithm. Although random mutations are computationally expensive, due to the necessity of checking the connectivity of zones, they are executed relatively few times. Selection for the next generation is straightforward. All these factors contribute to a fast convergence of the solution. The variance between different test runs is small. The exploration of the configuration space was done without a priori restrictions to the shapes of the clusters, employing a quantitative strategy to control its geometric irregularity. The elliptic scan is well suited for those clusters that fit well within some ellipse. The circular, elliptic, and SA scans have similar power in general. The elliptic scan method is computationally faster and is well suited for mildly irregular-shaped cluster

detection, but the non-compactness corrected SA and GA scans detects clusters with every possible shape, including the highly irregular ones. The choice of the statistic depends on the initial assumptions about the degree of shape irregularity to allow, and also on the availability of computer time.

The power of detection of the GA scan is similar to the simulated annealing algorithm for mildly irregular clusters and is slightly superior for the very irregular ones. The GA scan admits more flexibility in cluster shape than the elliptic and the circular scans, and its power of detection is only slightly inferior compared to these scans. The genetic algorithm is more computer-intensive when compared to the elliptic and the circular scans, but is faster than the simulated annealing scan. The use of penalty functions for the irregularity of cluster's shape enhances the flexibility of the algorithm and gives to the practitioner more insight of the geographic cluster delineation. We believe that our study encourages further investigations for the use of genetic algorithms for epidemiological studies and syndromic surveillance.

7. Acnowledgements

This work was partially supported by CNPq and CAPES.

8. References

Andrade LSS, Silva SA, Martelli CMT, Oliveira RM, Morais Neto OL, Siqueira Júnior JB, Melo LK, Di Fábio JL, 2004. Population-based surveillance of pediatric pneumonia: use of spatial analysis in an urban area of Central Brazil. Cadernos de Saúde Pública, 20(2), 411-421.

Assunçao R, Costa M, Tavares A, Ferreira S, 2006. Fast detection of arbitrarily shaped disease clusters. Statistics in Medicine 25;1-723-742.

Carrano, E.G., Soares, L.A.E, Takahashi, R.H.C, Saldanha, R.R., and Neto, O.M., 2006. Electric Distribution Network Multiobjective Design using a Problem-Specific Genetic Algorithm, *IEEE Transactions on Power Delivery*, 21, 995–1005.

Ceccato V, 2005 Homicide in São Paulo, Brazil: Assessing a spatial-temporal and weather variations. Journal of Enviromental Psychology, 25, 307-321

Conley J, Gahegan M, Macgill J, 2005. A genetic approach to detecting clusters in point-data sets. Geographical Analysis, 37, 286-314.

Duczmal L, Assunção R, 2004. A simulated annealing strategy for the detection of arbitrarily shaped spatial clusters, Comp. Stat. & Data Anal., 45, 269-286.

Duczmal L, Buckeridge DL., 2005. Using modified Spatial Scan Statistic to Improve Detection of Disease Outbreak When Exposure Occurs in Workplace – Virginia, 2004. *Morbidity and Mortality Weekly Report*, Vol.54 Suppl.187.

Duczmal L, Buckeridge DL, 2006a. A Workflow Spatial Scan Statistic. Stat. Med., 25; 743-754.

Duczmal L, Kulldorff M, Huang L., 2006b. Evaluation of spatial scan statistics for irregularly shaped clusters. J. Comput. Graph. Stat. 15:2;1-15.

Duczmal, L., Cançado, A.L.F., Takahashi, R.H.C., and Bessegato, L.F., 2007, A Genetic Algorithm for Irregularly Shaped Spatial Scan Statistics, *Computational Statistics and Data Analysis* 52, 43– 52.

Heffernan R, Mostashari F, Das D, Karpati A, Kulldorff M, Weiss D, 2004. Syndromic surveillance in public health practice, New York City. Emerging Infectious Diseases, 10:858

Iyengar, VS, 2004. Space-time Clusters with flexible shapes. IBM Research Report RC23398 (W0408-068) August 13, 2004.

Kulldorff M, Nagarwalla N, 1995. Spatial disease clusters: detection and inference. *Statistics in Medicine*, 14, 779-810.

Kulldorff M, 1997. A Spatial Scan Statistic, *Comm. Statist. Theory Meth.*, 26(6), 1481-1496.

Kulldorff M, 1999. Spatial scan statistics: Models, calculations and applications. In *Scan Statistics and Applications*, Glaz and Balakrishnan (eds.). Boston: Birkhauser, 303-322.

Kulldorff M, Tango T, Park PJ., 2003. Power comparisons for disease clustering sets, *Comp. Stat. & Data Anal.*, 42, 665-684.

Kulldorff M, Mostashari F, Duczmal L, Yih K, Kleinman K, Platt R., 2007, Multivariate Scan Statistics for Disease Surveillance. *Stat. Med.* (to appear).

Kulldorff M, Huang L, Pickle L, Duczmal L, 2006. An Elliptic Spatial Scan Statistic. *Stat. Med.* (to appear).

Lawson A., Biggeri A., Böhning D. *Disease mapping and risk assessment for public health*. New York, John Wiley and Sons, 1999.

Neill DB, Moore AW, Cooper GF, 2006. A Bayesian Spatial Scan Statistic. *Adv. Neural Inf.Proc.Sys. 18(in press)*

Patil GP, Taillie C, 2004. Upper level set scan statistic for detecting arbitrarily shaped hotspots. *Envir. Ecol. Stat.*, 11, 183-197.

Sahajpal R., Ramaraju G. V., Bhatt V., 2004 Applying niching genetic algorithms for multiple cluster discovery in spatial analysis. *Int. Conf. Intelligent Sensing and Information Processing*.

de Souza Jr. GL, 2005. Underreporting of Breast Cancer: A Study of Spatial Clusters in São Paulo State, Brazil. *M.Sc. Dissertation, Statistics Dept., Univ. Fed. Minas Gerais, Brazil.*

Takahashi RHC, Vasconcelos JA, Ramirez JA, Krahenbuhl L, 2003. A multiobjective methodology for evaluating genetic operators. *IEEE Transactions on Magnetics*, 39(3), 1321-1324.

Tango T, Takahashi K., 2005. A flexibly shaped spatial scan statistic for detecting clusters. *Int. J Health Geogr.*, 4:11.

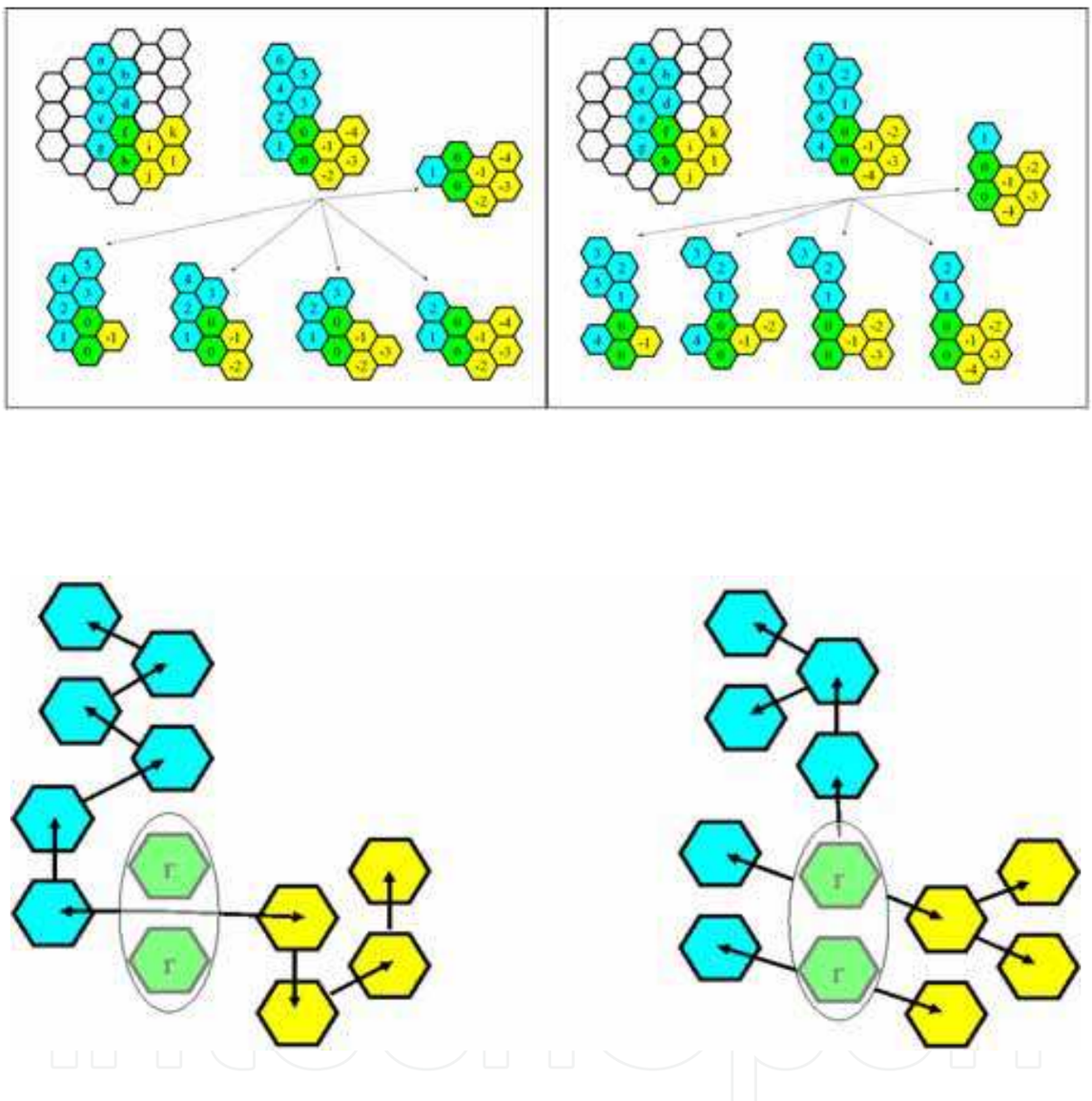


Figure 1A. The parents A ={a,*b,c,e,f,g,h*} and B ={*f,h,i,j,k,l*} have a common part C ={*f,g*}. Two possible level assignments are shown with their respective sets of trees. The level assignment to the left produces more regularly shaped offspring clusters, compared to the level assignment to the right of the figure.

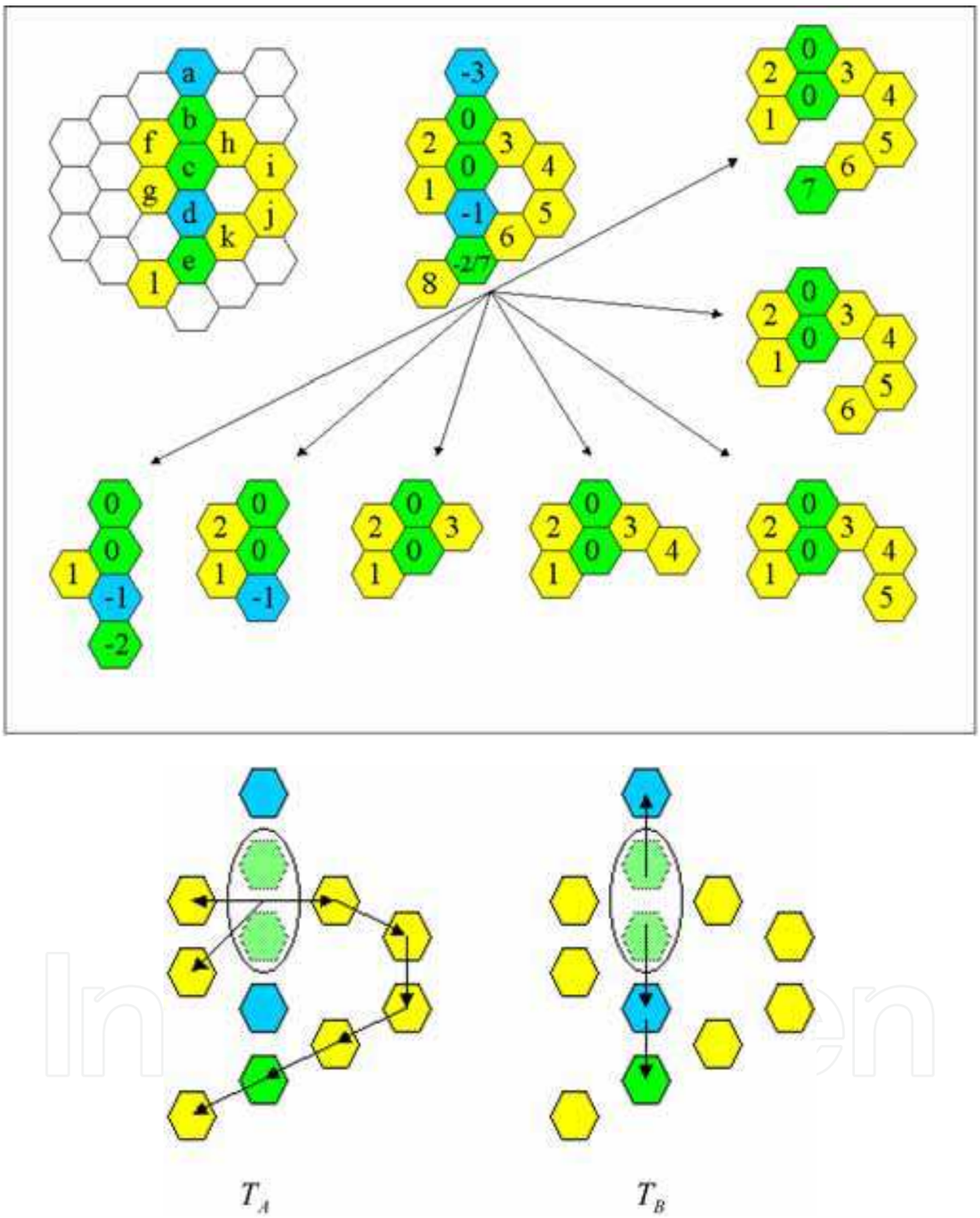


Figure 1B. The parents A ={*b,c,e,f,g,h,i,j,k,l*} and B ={*a,b,c,d,e*} have a common par C ={*b,c,e*}. In this example we choose the maximal connected set D ={*b,c*}. Observe that the node *e,* belonging to the set *C-D,* has both positive (7) and negative (-2) levels. The virtual root node *r* is made collapsing the two nodes of D (represented by the ellipse), and forms the root of the trees T_A (bottom left) and T_B (bottom right).

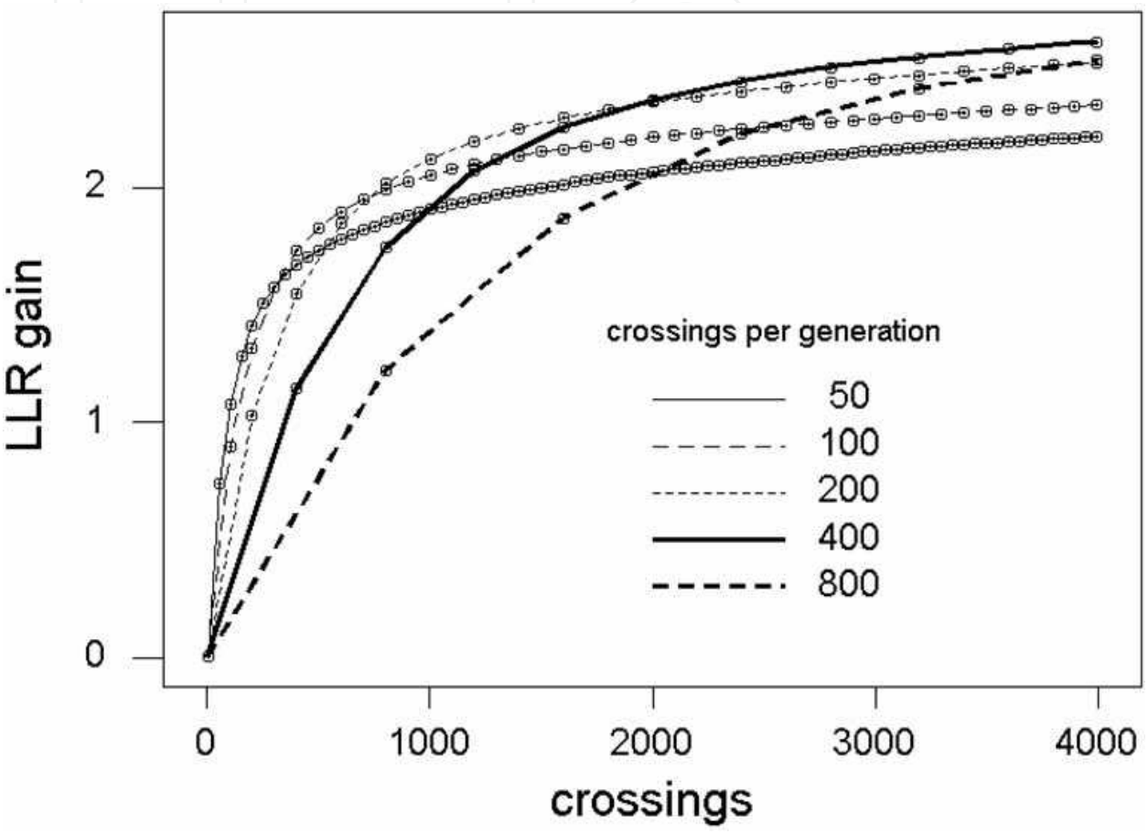


Figure 2. A numerical experiment shows how the number of wel succeeded crossings per generation (wsc_{MAX}) affects the LLR gain. Each little square, representing one generation, consists of the average of 5,000 runs of the genetic algorithm. A total of 4,000 wellsucceeded crossings were simulated for each run, for several values of wsc_{MAX}. In a given curve, with a fixed number of crossings per generation, the LLR value increases rapidly at the beginning, slowing further in the next generations. The optimal value for wsc_{MAX} is 400, in this case. Had the total of well-succeeded crossings been 1,000, the optimal value of wsc_{MAX} should be 200, as may be seen placing a vertical line at the 1,000 position.

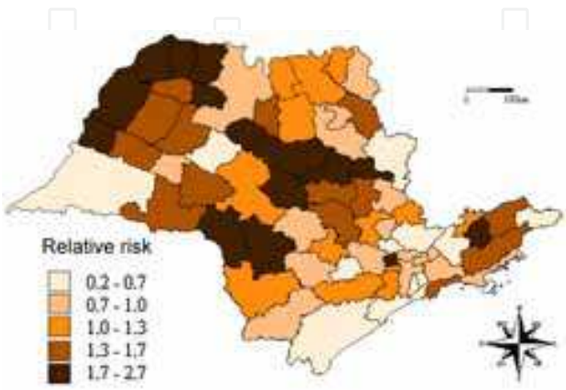


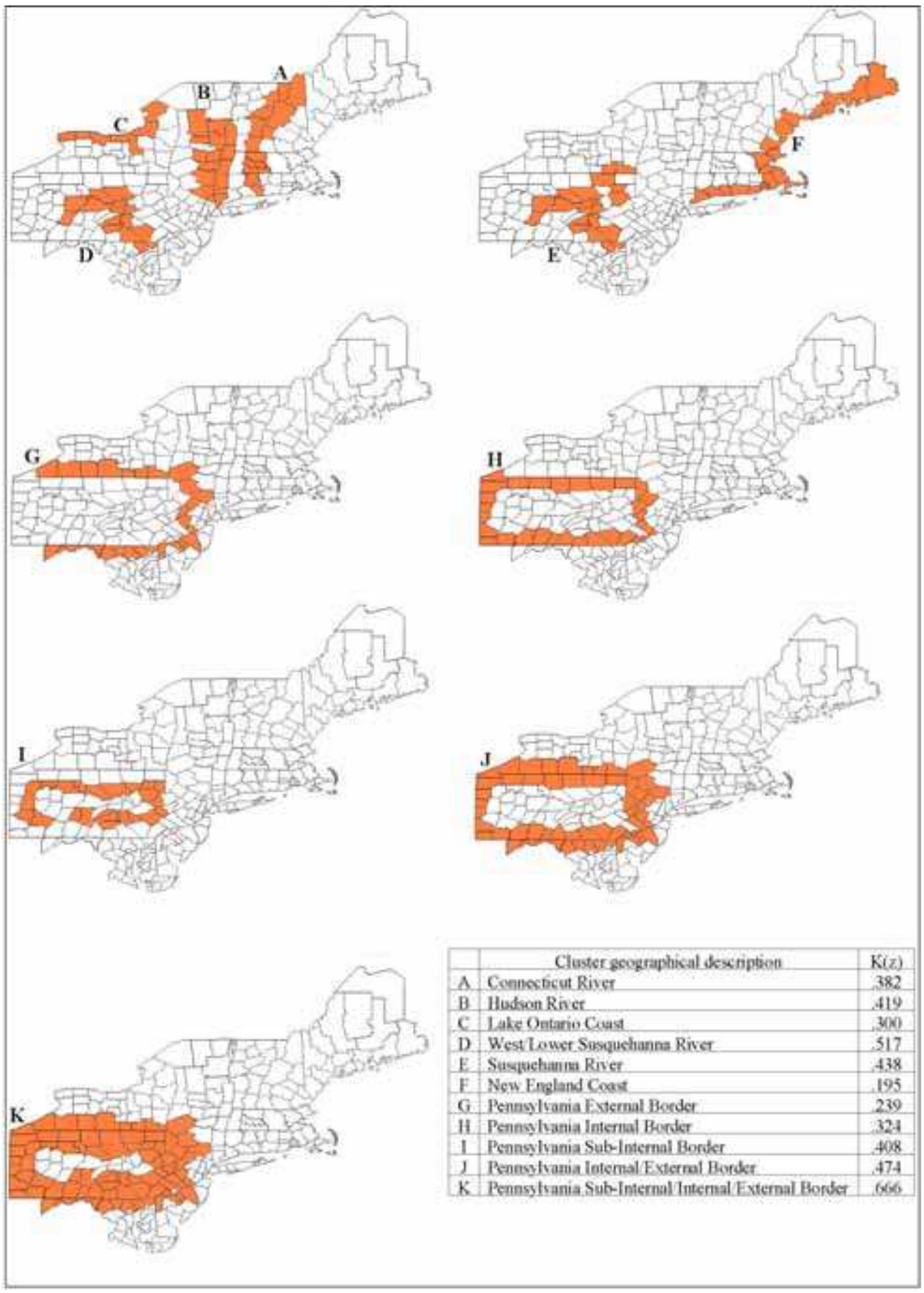
Figure 3A

Figure 3B

Figure 3C

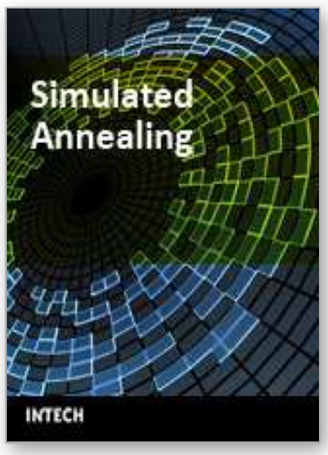
Figure 3D

Figure 3: The clusters of high incidence of breast cancer in São Paulo State, Brazil, during the years 2000-2003, found by the genetic algorithm. The map in Figure 3A displays the relative incidence of cases in each region. The maps 3B, 3C and 3D show respectively the clusters with penalty parameters a=1, a=0.5, and a=0. The primary (right) and secondary (left) circular clusters found by SatScan are indicated by the two circles in Figure 3D, for comparison.



	Cluster geographical description	K(z)
A	Connecticut River	.382
B	Hudson River	.419
C	Lake Ontario Coast	.300
D	West/Lower Susquehanna River	.517
E	Susquehanna River	.438
F	New England Coast	.195
G	Pennsylvania External Border	.239
H	Pennsylvania Internal Border	.324
I	Pennsylvania Sub-Internal Border	.408
J	Pennsylvania Internal/External Border	.474
K	Pennsylvania Sub-Internal/Internal/External Border	.666

Figure 4. New England's benchmark artificial irregularly shaped clusters used in the power evaluations.



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:

Luiz Duczmal, André L. F. Cançado, Ricardo H. C. Takahashi and Lupércio F. Bessegato (2008). A Comparison of Simulated Annealing, Elliptic and Genetic Algorithms for Finding Irregularly Shaped Spatial Clusters, Simulated Annealing, Cher Ming Tan (Ed.), ISBN: 978-953-7619-07-7, InTech, Available from: http://www.intechopen.com/books/simulated_annealing/a_comparison_of_simulated_annealing__elliptic_and_genetic_algorithms_for_finding_irregularly_shaped_

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