Available online at www.sciencedirect.com
ScienceDirect
Computational Statistics & Data Analysis 52 (2007) 43–52
COMPUTATIONAL STATISTICS & DATA ANALYSIS
www.elsevier.com/locate/csda

A genetic algorithm for irregularly shaped spatial scan statistics

Luiz Duczmal[a,*], André L.F. Cançado[b], Ricardo H.C. Takahashi[c], Lupércio F. Bessegato[a]

[a] *Statistics Department, Universidade Federal de Minas Gerais, Campus Pampulha, Belo Horizonte, MG 31270-901, Brazil*
[b] *Electrical Engineering Department, Universidade Federal de Minas Gerais, Brazil*
[c] *Mathematics Department, Universidade Federal de Minas Gerais, Brazil*

Available online 1 February 2007

Abstract

A new approach is presented for the detection and inference of irregularly shaped spatial clusters, using a genetic algorithm. Given a map divided into regions with corresponding populations at risk and cases, the graph-related operations are minimized by means of a fast offspring generation and efficient evaluation of Kuldorff's spatial scan statistic. A penalty function based on the geometric non-compactness concept is employed to avoid excessive irregularity of cluster geometric shape. The algorithm is an order of magnitude faster and exhibits less variance compared to the simulated annealing scan, and is more flexible than the elliptic scan. It has about the same power of detection as the simulated annealing scan for mildly irregular clusters and is superior for the very irregular ones. An application to breast cancer clusters in Brazil is discussed.

Keywords: Power evaluation; Genetic algorithm; Non-compactness penalty; Spatial scan statistic

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 et al., 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, 2006; Kulldorff et al., 2006, 2007) 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.

* Corresponding author. Tel.: +55 31 3499 5930; fax: +55 31 3499 5924.
E-mail address: duczmal@est.ufmg.br (L. Duczmal).

0167-9473/$ - see front matter © 2007 Elsevier B.V. All rights reserved.
doi:10.1016/j.csda.2007.01.016

Other methods, also using the scan statistic, were proposed recently to detect connected clusters of irregular shape (Duczmal and Assunção, 2004; Duczmal et al., 2006; Iyengar, 2004; Tango and Takahashi, 2005; Assunção et al., 2006; Neill et al., 2007). Patil and Taillie (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 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 paper derived from the previous work on the simulated annealing scan (Duczmal and Assunção, 2004; Duczmal et al., 2006). That algorithm 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 and Assunção (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. (2006), 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., 2006). 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 and Assunção, 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 and 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. (2006) 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 develop and implement a novel genetic algorithm cluster detector that incorporates the desirable features discussed above. It uses the spatial scan statistic in a map divided into a finite number of regions, offering a strategy to control the irregularity of cluster shape, generalizing the strategy used in the elliptic scan, which controls the shape in a more limited way. The algorithm provides 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 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, $\mathrm{O}(Z) = (C - c_Z)/(C - \mu_Z)$ is the relative incidence outside Z, it can be shown that

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

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 $LR(Z)$.

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 is as $K(z)=4\pi A(z)/H(z)^2$. Compactness penalizes a shape that has small area compared to the area of its convex hull (Duczmal et al., 2006). The strength of the compactness measure, employed here as a penalty factor, may be varied through a parameter $a \geqslant 0$, 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.

3. The genetic algorithm approach

We approach the problem of finding the most likely cluster by a genetic algorithm (GA) specifically designed for dealing with this problem structure.

3.1. The general structure of the genetic algorithm

A GA is defined as any algorithm that is structured with a set of N current candidate-solution points (these points are called *individuals* and the set of points is called *population*) that are evolved via the *genetic operators* (stochastic rules that lead a current population in a next population). The basic genetic operators are the *mutation operator* (which introduces random perturbations in some individuals), the *crossover operator* (which combines the features of two individuals, generating two new ones) and the *selection* (which applies a probabilistic rule for deciding which individuals will be selected for composing the new population, with greater chances assigned to the best individuals). It is known that some GAs 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 GAs (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.

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 $-1, -2, -3, \ldots$ for the levels, instead of $1, 2, 3, \ldots$ (see the example in Fig. 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 Fig. 1 again).

We now construct the offspring of the parents A and B as follows. Let $m_A \geqslant 2$ and $m_B \geqslant 1$ be, respectively, the number of elements of the sets $A - D$ and $B - D$, and suppose, without loss of generality, that $m_A \geqslant m_B$. 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.*

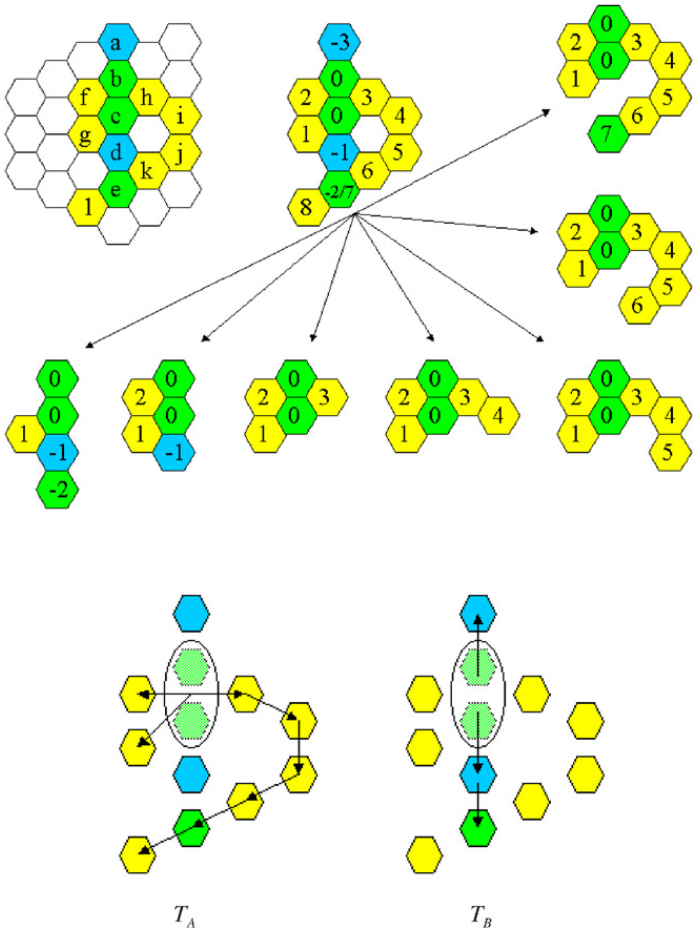

Fig. 1. The parents $A = \{b, c, e, f, g, h, i, j, k, l\}$ and $B = \{a, b, c, d, e\}$ have a common part $C = \{b, c, e\}$. In this example we choose the maximal connected set $D = \{b, c\}$. Observe that the node e, belonging to the set $C - D$, has both positive (7) and negative (-2) levels. The virtual 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).

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. □

In the example of Fig. 1, 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 Msets, 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.

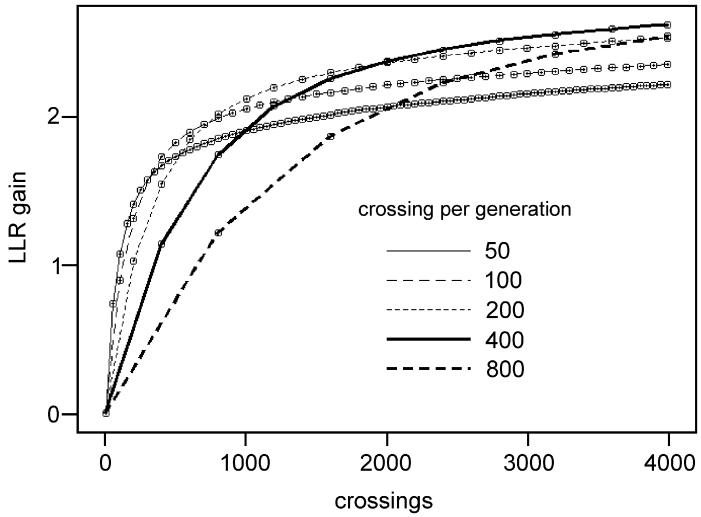

Fig. 2. A numerical experiment shows how the number of well-succeeded crossings per generation (wsc_{MAX}) affects the LLR gain. Each little square, representing one generation, consists of the average of 5000 runs of the genetic algorithm. A total of 4000 well-succeeded crossings were simulated for 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 1000, the optimal value of wsc_{MAX} should be 200, as may be seen placing a vertical line at the 1000 position.

We sort the current generation list in decreasing order by the LLR (modified as $\log(LR(z)^{K(z)^a})$ 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 LLR-ordered 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 Fig. 2 shows the results of numerical experiments. Each curve consists of the average of 5000 runs of the algorithm, varying wsc_{MAX} and G such that $wsc_{\mathrm{TOTAL}} = wsc_{\mathrm{MAX}} * G$, the total number of well-succeeded crossings, remains equal to 4000. 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 data set with real data population for the 245 counties Northeastern US map, with 11 simulated irregularly shaped clusters, that has been used in Duczmal et al. (2006). Clusters *A*–*E* are mildly irregularly shaped, in contrast to

Table 1
Power comparison between the genetic algorithm (GA) and the simulated annealing algorithm (SA), in parenthesis

Cluster	Size	Penalty	GA (SA) [8]	GA (SA) [12]	GA (SA) [20]	GA (SA) [30]
A	13	$a = 0$	0.84 (0.87)	0.84 (0.86)	0.79 (0.79)	0.68 (0.66)
		$a = 1$	0.85 (0.86)	0.85 (0.86)	0.84 (0.84)	0.80 (0.79)
B	16	$a = 0$	0.81 (0.83)	0.82 (0.84)	0.80 (0.81)	0.74 (0.74)
		$a = 1$	0.81 (0.78)	0.84 (0.84)	0.86 (0.86)	0.84 (0.83)
C	7	$a = 0$	0.87 (0.87)	0.86 (0.84)	0.82 (0.77)	0.72 (0.65)
		$a = 1$	0.80 (0.79)	0.78 (0.79)	0.74 (0.74)	0.68 (0.65)
D	15	a = 0	0.88 (0.89)	0.89 (0.90)	0.87 (0.88)	0.81 (0.81)
		$a = 1$	0.86 (0.85)	0.89 (0.89)	0.90 (0.90)	0.87 (0.87)
E	21	$a = 0$	0.83 (0.82)	0.86 (0.85)	0.87 (0.87)	0.84 (0.84)
		$a = 1$	0.77 (0.72)	0.82 (0.81)	0.86 (0.86)	0.87 (0.85)
F	23	$a = 0$	0.54 (0.58)	0.58 (0.61)	0.57 (0.59)	0.50 (0.51)
		$a = 1$	0.45 (0.44)	0.46 (0.45)	0.48 (0.46)	0.44 (0.44)
G	26	$a = 0$	0.58 (0.61)	0.62 (0.63)	0.66 (0.62)	0.68 (0.59)
		$a = 1$	0.50 (0.49)	0.53 (0.52)	0.55 (0.52)	0.55 (0.50)
H	29	$a = 0$	0.66 (0.69)	0.67 (0.70)	0.70 (0.69)	0.69 (0.67)
		$a = 1$	0.64 (0.62)	0.66 (0.67)	0.67 (0.67)	0.64 (0.64)
I	23	$a = 0$	0.66 (0.65)	0.71 (0.67)	0.74 (0.69)	0.71 (0.67)
		$a = 1$	0.62 (0.59)	0.64 (0.64)	0.68 (0.66)	0.70 (0.65)
J	55	$a = 0$	0.58 (0.60)	0.64 (0.66)	0.69 (0.69)	0.72 (0.70)
		$a = 1$	0.56 (0.54)	0.62 (0.63)	0.68 (0.67)	0.68 (0.67)
K	78	$a = 0$	0.53 (0.51)	0.61 (0.60)	0.69 (0.68)	0.75 (0.72)
		$a = 1$	0.47 (0.43)	0.56 (0.55)	0.67 (0.66)	0.72 (0.71)

The non-compactness 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 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 GA and SA scan statistics. For each upper limit of the detected cluster size, with ($a = 1$) and without ($a = 0$) non-compactness 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. The higher power values occur generally when the maximum size allowed matches the true size of the simulated cluster.

The power values for the statistics analyzed here are very similar. 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 ($a = 0$).

Numerical experiments show that the GA scan is approximately 10 times faster, compared to the SA scan presented in Duczmal and Assunção (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 min with a Pentium 4 desktop PC. Using exactly the same input for 5000 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

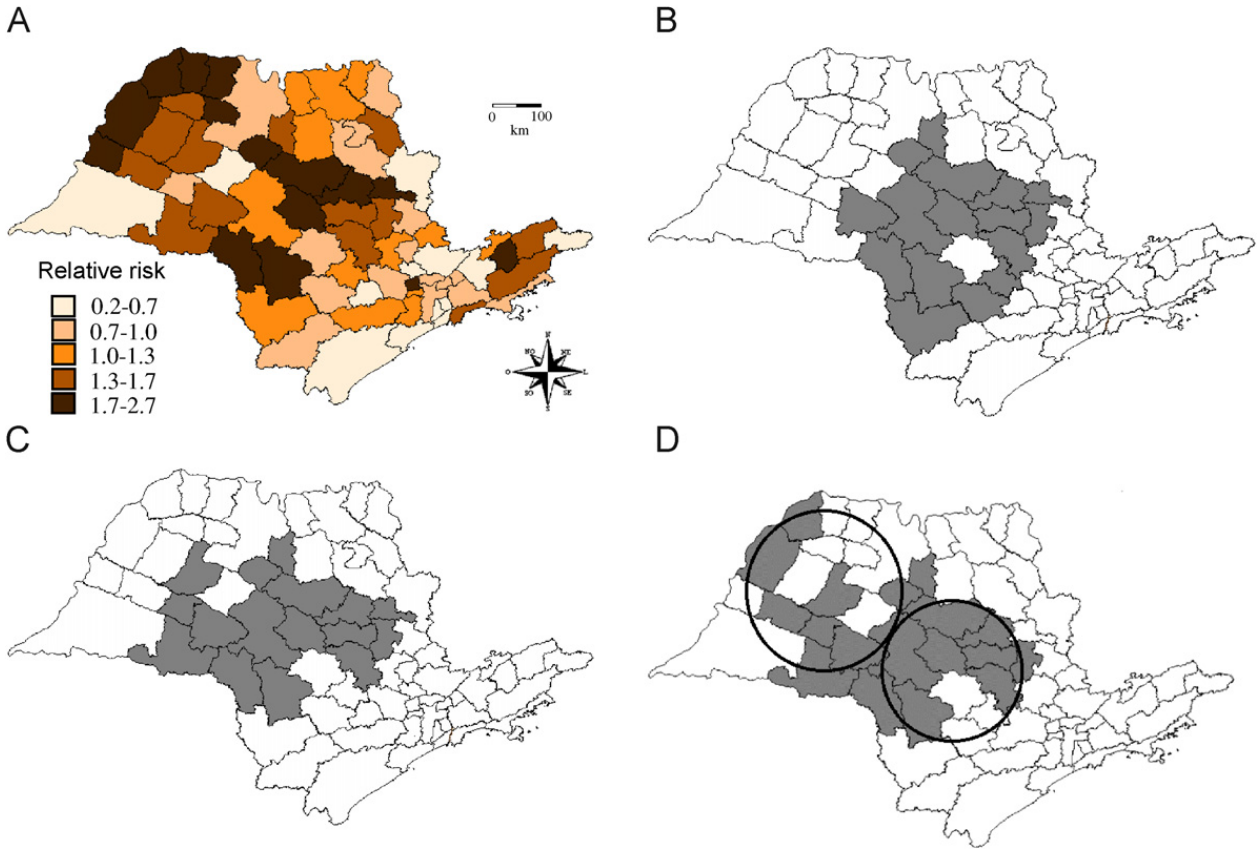

Fig. 3. The clusters of high incidence of breast cancer in São Paulo State, Brazil, during the years 2000–2003, found by the genetic algorithm. The map in Fig. 3A displays the relative incidence of cases in each region. The maps 3B, 3C and 3D show, respectively, the clusters with penalty parameters $a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

Table 2
The three clusters of Fig. 3B–D

Figure	A	Size	Cases	Population	Incidence	LLR	p-value
3B	1.0	16	3324	394,294	0.00843	298.9	0.001
3C	0.5	16	3078	361,373	0.00852	343.8	0.001
3D	0.0	18	2924	346,024	0.00845	449.6	0.001

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. Fig. 3A shows the relative incidence of cases for each region, where the darker shades indicate higher incidence of cases. The other three maps (Fig. 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 Fig. 3B should be compared with the primary circular cluster that was found by SatScan (the rightmost circle in Fig. 3D). It is also interesting to compare the cluster in Fig. 3D with the primary and secondary circular clusters that were found by the circular SatScan algorithm (see the circles in Fig. 3D).

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 and the 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 power of detection is similar to the simulated annealing algorithm for mildly irregular clusters and is slightly superior for the very irregular ones. The genetic algorithm 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.

Acknowledgments

We thank the editor, the referees, and Martin Kulldorff for their valuable comments. This work was partially supported by CNPq.

References

Andrade, L.S.S., Silva, S.A., Martelli, C.M.T., Oliveira, R.M., Morais Neto, O.L., Siqueira Júnior, J.B., Melo, L.K., Di Fábio, J.L., 2004. Population-based surveillance of pediatric pneumonia: use of spatial analysis in an urban area of Central Brazil. Cad. Saúde Pública 20 (2), 411–421.

Assunção, R., Costa, M., Tavares, A., Ferreira, S., 2006. Fast detection of arbitrarily shaped disease clusters. Statist. Med. 25, 723–742.

Carrano, E.G., Soares, L.A.E., Takahashi, R.H.C., Saldanha, R.R., Neto, O.M., 2006. Electric distribution network multiobjective design using a problem-specific genetic algorithm. IEEE Trans. Power Delivery 21 (2), 995-1005.

Ceccato, V., 2005. Homicide in São Paulo, Brazil: assessing a spatial–temporal and weather variations. J. Environ. Psychol. 25, 307–321.

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

Duczmal, L., Assunção, R., 2004. A simulated annealing strategy for the detection of arbitrarily shaped spatial clusters. Comput. Statist. Data Anal. 45, 269–286.

Duczmal, L., Buckeridge, D.L., 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, D.L., 2006. A workflow spatial scan statistic. Statist. Med. 25, 743–754.

Duczmal, L., Kulldorff, M., Huang, L., 2006. Evaluation of spatial scan statistics for irregularly shaped clusters. J. Comput. Graph. Statist. 15 (2), 1–15.

Heffernan, R., Mostashari, F., Das, D., Karpati, A., Kulldorff, M., Weiss, D., 2004. Syndromic surveillance in public health practice, New York city. Emerging Infect. Dis. 10, 858.

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

Kulldorff, M., 1997. A spatial scan statistic. Comm. Statist. Theory Methods 26 (6), 1481–1496.

Kulldorff, M., 1999. Spatial scan statistics: models, calculations and applications. In: Glaz, J., Balakrishnan, N. (Eds.), Scan Statistics and Applications. Birkhauser, Boston, pp. 303–322.

Kulldorff, M., Tango, T., Park, P.J., 2003. Power comparisons for disease clustering sets. Comput. Statist. Data Anal. 42, 665–684.

Kulldorff, M., Huang, L., Pickle, L., Duczmal, L., 2006. An elliptic spatial scan statistic. Statist. Med. 25, 3929–3943.

Kulldorff, M., Mostashari, F., Duczmal, L., Yih, K., Kleinman, K., Platt, R., 2007. Multivariate scan statistics for disease surveillance. Statist. Med., in press.

Lawson, A., Biggeri, A., Böhning, D., 1999. Disease Mapping and Risk Assessment for Public Health. Wiley, New York.

Neill, D.B., Moore, A.W., Cooper, G.F., 2007. A Bayesian spatial scan statistic. Adv. Neural Inf. Process. Syst., in press.

Patil, G.P., Taillie, C., 2004. Upper level set scan statistic for detecting arbitrarily shaped hotspots. Environ. Ecol. Statist. 11, 183–197.

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

de Souza, Jr., G.L., 2005. Underreporting of breast cancer: a study of spatial clusters in São Paulo State, Brazil. M.Sc. Dissertation, Statistics Department, Universidade Federal de Minas Gerais, Brazil.

Takahashi, R.H.C., Vasconcelos, J.A., Ramirez, J.A., Krahenbuhl, L., 2003. A multiobjective methodology for evaluating genetic operators. IEEE Trans. Magn. 39 (3), 1321–1324.

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