APPLICATION OF MAPPING PLAN WITH A NON-DETERMINISTIC ALGORITHM FOR GIS QUERYING

V. Barrile[a], F. Cotroneo[b], S. Tringali
DIMET, Università degli Studi "Mediterranea", Via Graziella Feo di Vito,
89100 Reggio Calabria, Italy – [a]barrile@ing.unirc.it; [b]fcotroneo@ing.unirc.it

KEY WORDS: GIS, Estimation, Query, Artificial Intelligence, Advancement

ABSTRACT

GIS systems are an irreplaceable tool for "decision supporting". In fact, one of the most important characteristic of a GIS is its ability to offer users and planners proper tools in order to question the same system, that is to make some query of different kind and retrieve useful data for any following decision. This is because the GIS reply can "make explicit" the informative contents, already existing in stored data but in a implicit way, because of the incorrelation among them. While concerning the spatial query, these querying tools are the overlay functions and union functions between layer and proximity, in reference to alphanumeric data GIS softwares use the SQL language to make the queries. Retrieved data allow a management and an advanced control of territory.
Nevertheless, there are some conditions in which it need to have informations that are unreachable with querying tools above analyzed: for example, to evaluate correlation between cartographic data and non-deterministic variables (i.e. data on environmental variations, social and economical variables, etc...). Because of these requirements, during last years some Artificial Intelligence (AI) based algorithms are implemented and embedded in GIS softwares, in order to allow the automatic or self-controlled creation from raster layers of thematic maps with equivalence classes hardly identifiable by classic methods (such as spectral analysis); other AI-based solutions are applied to an "on the fly" overlay between raster and vectorial layers.
The aim of our work is to illustrate a further possibility for usage of intelligent algorithms with GIS; in fact, we want to show how it is possible to implement new spatial querying operators (Overlay A.I., Union A.I., Proximity A.I.) by means of neural networks, with a redefinition of classical ones, and the possibility to obtain the "fuzzy" results above described in a typical application of heritage conservation and mapping planning sciences.

1. INTRODUCTION

The primary objective of this paper is to define a new set of operators, useful in order to extract a rich variety of information from a specific typology of geospatial databases, not otherwise obtainable by means of the classical operators employed in the softwares of current availment, which are necessary to build new layers by elaborating the related operands through algorithms of artificial intelligence.
To face this typology of problematics to the best, it becomes essential to insist on the introduction of a suitable formalism fit to describe the database of a GIS through a language free from any possible lexical ambiguity. It's just for this reason that the treatment ot the thematics we're going to face follows to a revisitation, in a formal key, of the typical definition related to the characteristic elements of a GIS. In this way, you make besides possibile to accomplish a comparison with the standard operators in a relatively simplified perspective, in reference both to the output data and to the adopted methodology of elaboration. Specifically, our study has gotten as its final result to place side by side to the default operators of intersection (overlay), union and proximity one custom version of theirs: the intersection A.I., the union A.I. and the proximity A.I.

2. AN ALTERNATIVE WAY TO SPEAK ABOUT GIS

In our meaning, a *layer* L is a subset of the cartesian product $S \times G$, in which $S = S_1 \times S_2 \times \ldots \times S_n$ is a set of alphanumeric data organized within n different fields to form (as it's usually said) a *record* and G is a set of geometric entities, each belonging to the same plane π, such that, if $(s_1, g_1), (s_2, g_2) \in L$ and $g_1 = g_2$, then $s_1 = s_2$ too. Of fact, if (s, g) is an arbitrary element of the layer, that's to say a *feature*, then s is an n-dimensional array whose i-th component consists of a sequence of symbols all belonging to the same alphabet, both proper of human langua-ges or typical of machine ones. In this sense, the set of all the records which appartain to the same layer can be usefully represented by a table consisting of n columns and m lines, where m is the layer cardinality, i.e. the number of its elements, always supposed to be finite, for practical requirements.
Then you assume that, for every $i = 1, 2, \ldots, n$, S_i is a structured set, in which you have eventually defined not only one or more internal operations like addition or multiplication, as in the typical case of numeric data, but also relational operators of a manifold kind (for example, operators of comparison).
With reference to the set of the geometric data, within this artiche we assume G may be specialized, in practice, into three fundamental typologies:

- the set Λ of all points;
- the set Σ of all open polygonals, or polylines;
- the set Γ of all not woven polygons.

In this sense, the GIS database is just the union set among k layers $L_1, L_2, \ldots, L_k$. Querying operations executed inside a GIS consist essentially of searches performed on the elements of one or more layers, and they are effected by a suitable composition of a whole range of primitive operators, already available among the base functions implemented in the maximum part of general purpose GIS softwares (for example, ArchiGIS, GRASS, MapInfo), declared to operate both over the geometric data in the layer - as in the case of a spatial intersection or when you may be interested in a buffering operation by proximity - or over the respective alphanumeric ones, in which case the search usually employes the typical structures of declarative languages (for example, SQL is a frequently used solution).

3. TOWARDS THE A.I. OPERATORS

From an abstract point view, an *n*-ary operator on layers is a function $\Phi(\cdot)$: $L_1 \times L_2 \times \ldots \times L_n \rightarrow L$, where $L_1, L_2, \ldots, L_n$ and L are layers related to the same GIS system. To state an example, standard intersection may be applied on two layers $L_1 \subseteq S_1 \times \Gamma$ and $L_2 \subseteq S_2 \times \Gamma$ as an operator $int(\cdot)$: $L_1 \times L_2 \rightarrow L$, in which, if $(s, g) \in \mathrm{Im}\ int(\cdot) \subseteq L$, then $s = (s_1 : s_2)$ and $g = g_1 \cup g_2$, where $(s_1, g_1) \in L_1$ and $(s_2, g_2) \in L_2$ (see figure 1).

Here and so in the following, we assume $(s_1 : s_2)$ denotes the operation of merging between the *m*-tuple s_1 and the *n*-tuple s_2, if $S_1 = S_{11} \times S_{12} \times \ldots \times S_{1m}$ and $S_2 = S_{21} \times S_{22} \times \ldots \times S_{2m}$.

This not standing, within the targets proper of this article, we're going to consider a special class of generalized operators in the form $\Phi(\cdot)$: $L \times F_{1L} \rightarrow F_{2L}$, where $L = \{(s_1, g_1), \ldots, (s_n, g_n)\}$ is supposed to represent a layer of points, polylines or polygons; $F_{1L} = \{L_{11}, L_{12}, \ldots, L_{1r}\}$ and $F_{2L} = \{L_{21}, L_{22}, \ldots, L_{2s}\}$, on the other hand, are two families of layers, each being characterized by a specific geometric typology.

In practice, the key concept is that our generalized operators act mapping every couple of operands not into single elements of a layer, but into a whole layer, in a way which is quite different from the correspondence classically ensured. Specifically, we're fixing firstly our attention on the A.I. intersection operator.

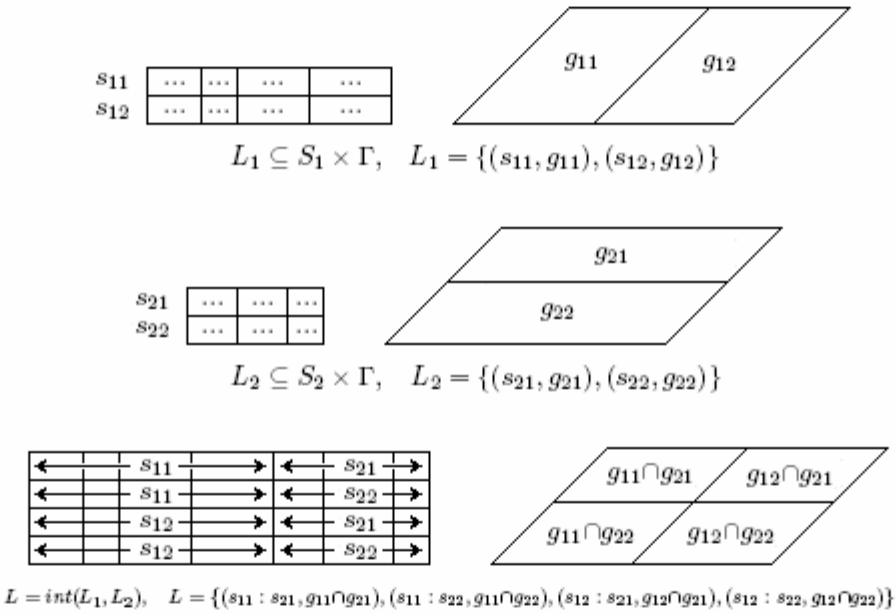

Figure 1. A representation illustrating the behaviour of standard intersection regarded as an operator.

4. THE A.I. INTERSECTION

In our meaning, the A.I. intersection is definied in terms of an operator $int_{A.I.}(\cdot)$: $L_P \times F_{1L} \rightarrow F_{2L}$, where $L_P \subseteq S_P \times \Lambda$ is a layer of points, intended as *sources* of a certain typology of propagating signal; F_{1L} and F_{2L} are classes of polygonals layers, in which $L_1 \subseteq S_1 \times \Gamma$ and $L_2 \subseteq S_2 \times \Gamma$, if $L_1 \in F_{1L}$ and $L_2 \in F_{2L}$, given that Λ and Γ are referred to a same geometric plane π.

Without any necessity of being too specific, we assume that S_1 contains at least the fields necessary for any processing involved in the definition of the A.I. intersection.

To make this aspect clear, we say the single polygon within the image-layer of our operator is built employing a neural network which, scanning the time at successive instants $t_1, t_2, \ldots, t_m$, for a certain $m \in$ (all computed starting from $t_0 = 0$ sec), estimates distances $r_j^1, r_j^2, \ldots, r_j^n$, expressed in a suitable metric unity u, reached by the advanced front of an impulsive signal of assigned magnitude, generated at the source-point P and proceeding along the half-line traced from P in the direction pointed by the angle $\theta_j = 2j\pi/N$ (where $j = 0, 1, \ldots, N-1$), which results from the anticlockwise splitting of a full angle into N equal parts, with respect to a polar reference system centered in P. Of fact, the generic r_j^i is calculated according to the characteristic recursive relation:

$$r_j^{i+1} = r_j^i + f(n, t_{n+1}, t_n, \ldots, t_1, r_j^i, P, L), \qquad [1]$$

where $r_j^0 = 0\ u$; $f(\cdot)$ is a multivariable function valued in $^{+}$ and definied on a certain discrete subset Ω of an euclidean space of type ^{p}, for a suitable $p \in$; $P \in L_P$ and $L \in F_{1L}$ identify the operands mapped by $int_{A.I.}(\cdot)$, which are here intended as argument of $f(\cdot)$ to represent generically all the alphanumeric and geometric data effectively involved in the elaboration accomplished by the neural network. For example, with respect to the applications to the study of a propagating seismic signal of which you know the source position and the amplitude of the mechanical impulse emitted at the instant $t_0 = 0$ sec, we shall consider the relation $r_j^{i+1} = r_j^i + (t_n - t_{n-1}) \cdot v_j^{i+1}$, where v_j^{i+1} represents the speed which is proper of the signal while it's running from the point r_j^i to the adjacent point r_j^{i+1} along the direction addressed by the angle θ_j, for every $i = 0, 1, \ldots, n$. and $j = 0, 1, \ldots, N-1$. This speed is calculated employing the alphanumeric and geometric data included within the polygonal layer passed as an argument for the A.I. intersection operator and specifically containing sampled punctual informations about the PSD (power spectral density) of the geological bed.

Once completed the phase of learning, the neural network which the system of processing interfaces itself to is capable to extrapolate the PSD of every point in the geological bed starting from the sampled values stored inside the GIS, repeating this procedure for every layer $L \in F_{1L}$. So you are guaranteed to obtain a suitable approximation of the signal propagating speed at *any* point of the geological bed, even if this doesn't belong to the our sampling set. Hence it's necessary to admit, as principle, every $L \in F_1(L)$ has one or more fields describing by samples (ideally with continuity) a class of parameters on which the propagation of the seismic signal is dependent. So, if $((s, g), L)$

represents the generic operand passed as an argument to $int_{A.I.}(\cdot)$ and we put Im $int_{A.I.}(\cdot) = \{L_{01}, L_{02}, \dots, L_{0r}\} \subseteq F_{2L}$, under the hypothesis that r is the number of sources included within the punctual layer L_P, we find the generical feature $(s_{ik}, g_{ik}) \in L_{0k}$, for any $k = 1, 2, \dots, r$, is such that: g_{ik} is the polygon having as internal and external frontiers, respectively, the two closed polygonals obtained linking by a segment all the adjacent points $r_0^{i}, r_1^{i}, \dots, r_{N-1}^{i}$ and $r_0^{i+1}, r_1^{i+1}, \dots, r_{N-1}^{i+1}$ traced starting from the source point P_k and stored within the processing system including our neural network into a sort of matrix structure, as illustred in figure 2; and s_{ik} is a 3-tuple $(s_{ik}^1, s_{ik}^2, s_{ik}^3)$ where s_{ik}^1 and s_{ik}^2 save, respectively, the signal amplitude on the internal and external frontier of g_{ik}, and s_{ik}^3 reports the absolute coordinates of the source point P_k.

	t_1	t_2	$\dots$	t_n
$\theta_0 = 0$	$r_0^1 = t_1 v_0^1$	$r_0^2 = r_0^1 + (t_2 - t_1) v_0^2$	$\dots$	$r_0^n = r_0^{n-1} + (t_n - t_{n-1}) v_0^n$
$\theta_1 = 2\pi/m$	$r_1^1 = t_1 v_1^1$	$r_1^2 = r_1^1 + (t_2 - t_1) v_1^2$	$\dots$	$r_1^n = r_1^{n-1} + (t_n - t_{n-1}) v_1^n$
$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$
$\theta_{m-1} = 2(m-1)\pi/m$	$r_{m-1}^1 = t_1 v_{m-1}^1$	$r_{m-1}^2 = r_{m-1}^1 + (t_2 - t_1) v_{m-1}^2$	$\dots$	$r_{m-1}^n = r_{m-1}^{n-1} + (t_n - t_{n-1}) v_{m-1}^n$

Figure 2. Display of the matrix structure stored inside the processing system and used to build the geometric data of $int_{A.I.}(\cdot)$.

5. THE A.I. UNION

Likewise, the A.I. union is defined in terms of an operator $cup_{A.I.}(\cdot)$: $L_1 \times L_2 \rightarrow L$, in which L is a polygonal layer and L_1 and L_2 are distinct elements in the image of the A.I. intersection operator, which we are supposing characterized by a cardinality $r > 1$. In practice, $cup_{A.I.}(\cdot)$ takes two features $(s_1, g_1) \in L_1$ and $(s_2, g_2) \in L_2$, where L_1 and L_2 are "generated", respectively, by impulsive signals propagating from the source points P_1 e P_2; it carries out the intersection set on the associated geometric data and splits $g_1 \cup g_2$, supposed not empty, into a certain number k of polygons $\gamma_1, \gamma_2, \dots, \gamma_k$, obtained still determining the geometric intersection between $g_1 \cup g_2$ and any couple of angles of assigned and constant amplitude θ, whose vertices are placed in P_1 and P_2. The centroid C_i of every γ_i is subsequently calculated in order to allow the processing system whose core is our neural network to establish the direction and intensity of the signal propagating from sources P_1 and P_2 to the point C_i (see figure 3). The vectorial data reconstructed by this means are finally added to generate the output couple (s_{0i}, g_{0i}), in which $g_{0i} = \gamma_i$ and s_{0i} gives the intensity of the overlapped signal.

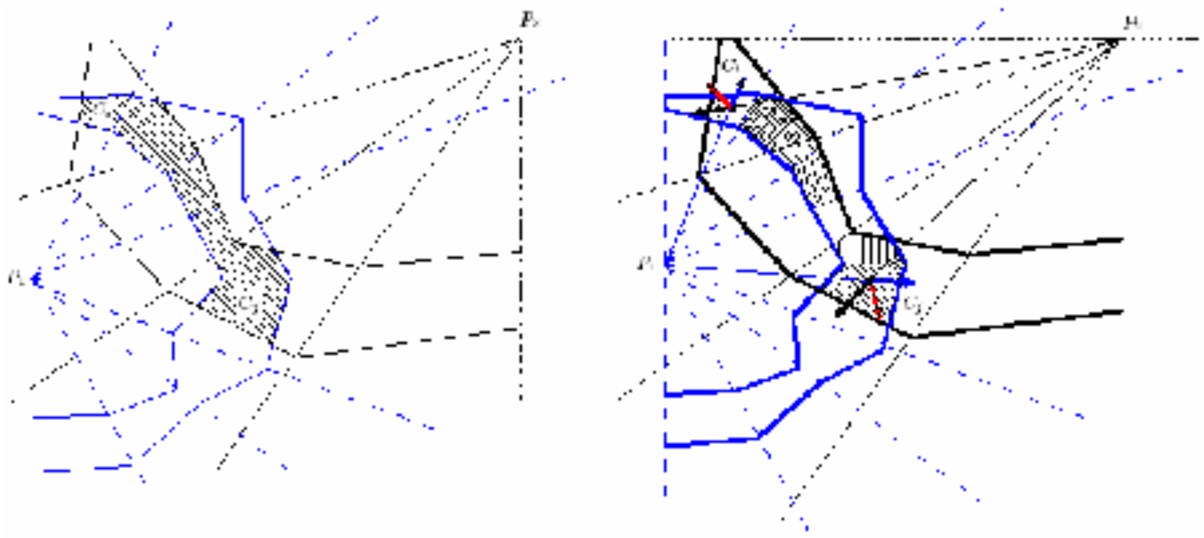

Figure 3. Two steps explaining the operation accomplished by the A.I. union.

6. THE A.I. BUFFERING

The A.I. proximity is defined as an operator $prox_{A.I.}(\cdot)$ which accomplishes the standard intersection between an element of an arbitrary layer $L \subseteq S \times G$, where $G \in \{\Lambda, \Sigma, \Gamma\}$, and the standard union of all the polygonal features $((s_i^1, s_i^2, s_i^3), g)$, belonging to a layer L_i in the image of $int_{A.I.}(\cdot)$, such that s_i^2 is lower than a buffering value setted by users.

7. CONCLUSIONS

A.I. intersection, union and proximity operators definied within this paper allow to obtain informations useful to understand the time evolution of a certain propagating phenomenon inside an area which effects its properties. Of course, such operators are not intended to substitute the numerous methods of numerical analysis already avaiable in literary, but they results particularly useful when you want to get a qualitative representation of the phenomenon itself, or when the addressed numeric methods can't be applied because of a scarse amount of useful data.

REFERENCES

Fu, L. M. 1994. Neural Networks in Computer Intelligence. New York: McGraw-Hill, pp. 101-130.

Simpson, Patrick K. 1990. Artificial Neural Systems : Foundations, Paradigms, Applications, and Implementations. New York: Pergamon Press, pp. 80-103.

Paul A. Longley and Michael Batty. 2003. Advanced Spatial Analysis: The CASA book of GIS, Editors ESRI Press, pp. 340-368.

Christos Christodoulou, Michael Georgiopoulos. 2000. Applications of Neural Network in Electromagnetic, Editors Artech House Publishers pp. 315-372.

Valeriano Comincioni. 1995. Analisi Numerica. Editors McGRAW-HILL, pp. 753 - 793.

Daniel A. Griffith, Larry J. Layne, J. K. Ord, Akio Sone. 1996. A Casebook for Spatial Statistical Data Analysis: A Compilation of Analyses of Different Thematic Data Sets (Spatial Information Systems), pp. 60 – 73.