Dynamic Datastructures

THE INTERACTIVE MAP

CHRISTOPHER M. GOLD,
Chaire Industrielle en Géomatique,
Centre de Recherche en Géomatique,
Pavillon Casault, Université Laval,
Québec, Qc Canada G1K 7P4.
Telephone: (418) 656-3308,
Internet: CGOLD@VM1.ULAVAL.CA

ABSTRACT

It is possible to eliminate some of the traditional problems with GIS implementation by changing the model of space used. The Voronoi spatial model has bee used as the basis of a set of core GIS or spatial operations, with encouraging results. It has been implemented in a PC environment so as to leave as much flexibility as possible for the designer of any particular application. Examples include polygon dissolve, buffer zone, overlay, interpolation and robot navigation problems.

1. INTRODUCTION: INTERACTIVE DSS AND THE SPATIAL MODEL.

Approaches to spatial data modelling and queries have often been limited by the model of space being used, the batch nature of spatial relationship generation, and the subsequent static view of spatial data. Due to the absence of true graphical interaction of the user with the "map" (e.g. via a cursor or mouse) spatial queries have also been considered to be static operations.

In the effort to achieve a basic interactive spatial decision support system, in this case for forestry applications, it became necessary to re-evaluate the assumptions on which the traditional raster and vector spatial models were based - and in particular the definition and preservation of the adjacency relationships between objects. In brief, the raster model does preserve adjacency relationships through the implicit tiling of the map space, but does not relate directly to objects embedded in that space - thus it is more useful for representing field information, e.g. elevation or temperature. The vector model assumes a polygonal tiling of space, but it must be constructed by the laborious specification of boundaries and nodes. It may thus be used to specify objects, but their spatial relationships are dependent on the detection of line intersections. The vector model thus is not appropriate for spatially unconnected or non-polygonal features, and forms a functional topology only after all errors are corrected and additional containment relations are specified.

2. ADVANTAGES OF THE VORONOI SPATIAL MODEL.

The Voronoi model (see the survey in Aurenhammer 1991) has the complete tiling property of the raster model, but the tiles are based on the proximal region around each object (e.g. a point or a line segment in the current implementation, as in Figure 1). Static methods for constructing the Voronoi diagram for both points and line segments are known (e.g. Fortune, 1986), but fully dynamic local methods are new. (Roos, 1990 has produced point movement algorithms.) Here the unambiguous definition of spatial adjacency as being a property of two polygons with a common boundary may be preserved under object movement, insertion or deletion, as the Voronoi tessellation is space-filling.

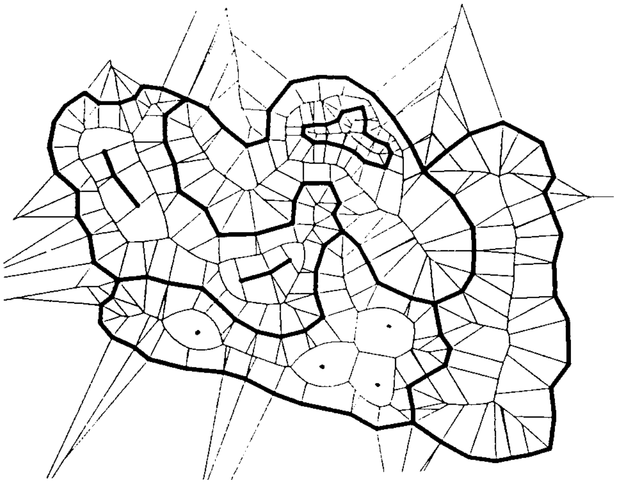


Figure 1. The Voronoi tessellation of a set of points, line segments and polygons.

2.1 Local updating

This gives a fully local means of preserving "topology", allowing fully interactive map update in response to the user's graphical actions (see Gold, 1992). It also permits the mixing of many types of data - fully connected polygons, connected hydrography, discrete points and line segments - within the same overlay. Each layer is the adjacency specification of a selected set of objects (expressed as the dual Delaunay triangulation, with pointers between adjacent triangles, and to the objects forming triangle vertices). The preservation of the Voronoi structure as a point moves is achieved by detecting "topological events" (Roos, 1990) and then switching triangle pairs.

2.2 Multiple layers, single objects

There are three record types stored in the Voronoi system: coordinate records (containing x,y, etc.); Delaunay triangles (containing pointers to the three adjacent triangles and to the three map objects forming the vertices); and the map objects themselves, which may be points or half-line segments (and have pointers, as required, to the coordinate records and the matching half-line). As there are no direct pointers associated with map objects, (all topology being contained within the Delaunay triangle records) the same object may be inserted into several layers simultaneously. Thus a map layer consists of a pointer to any triangle within a particular mesh that may be followed to the desired location or map object. Different layers, being defined solely as the set of neighbourhood relationships (triangles), may include the same map object. This eliminates object duplication in the attribute database, as well as differing coordinate representations of the same object in various layers - eliminating the consequent sliver polygons when overlaid.

2.3 Representation of fields and objects simultaneously

Because all objects have a proximal zone, the "area-stealing" interpolation model may be used to estimate field values (e.g. elevation) at intermediate locations (see Gold, 1989). These map objects may be of any form, thus permitting precise interpolation between points, line segments or complete polygons, with any specified level of continuity, eliminating the distinction between object and field data. Figure 2 shows interpolation to a line segment.

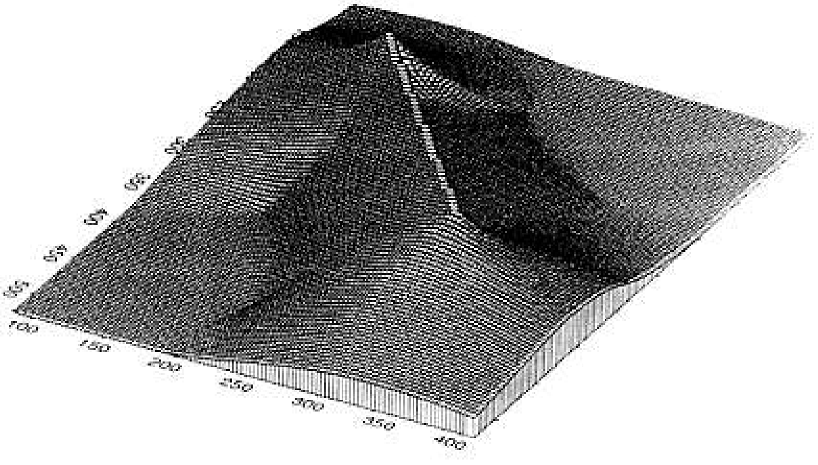


Figure 2. Interpolation to a single "high" line segment surrounded by low data points.

2.4 Choice of coordinate systems

The simple (nearest-object) Voronoi diagram may be implemented in various dimensions and metrics. While currently only implemented in two dimensions in the Euclidean metric, the same approach may be used in various others. Work is in progress to implement the operations described here on the sphere.

2.5 The interactive map for interactive decision support, and its interface

The underlying design described here has permitted the development of prototypes of new interactive spatial decision support tools. The dynamic spatial data structure supports incremental construction and editing of the map. This has the particular advantage that in specifying previously-defined objects to connect, for example, it is necessary merely to select them by "pointing" within their Voronoi zones, followed by the appropriate command mapped to the cursor buttons. (This "proximal query" is basic to all interaction, and is equivalent to normal human gestures for specifying objects.)

2.6 Incremental actions and recorded history

A map is built up of a sequence of incremental actions, as with a paper map, and may subsequently be updated in the same fashion by adding or deleting points or line segments. This allows the creation of a historical "tape recording" of the map construction that may be replayed to any desired date. This gives a basic level of temporal query - the whole map may be played forwards or backwards to a particular date, or a particular point location may be watched and have all state changes reported. Further research is under way to expand these forms of spatio-temporal query.

3. SYSTEM DESIGN

3.1 The object-oriented approach to spatial objects

All construction commands are naturally "object-oriented" in that an object is selected and the appropriate command passed to it as a message. There are nine such commands (three ways of splitting a new point from an old and three merge options, along with point movement - see Gold 1992). For example, we may move an existing point (no split, no merge), or create a new line segment (split a new point and trailing line segment from a pre-existing point, and then move the point to the desired location). Deletion follows the reverse sequence from creation. Figure 3 illustrates the nine basic commands.

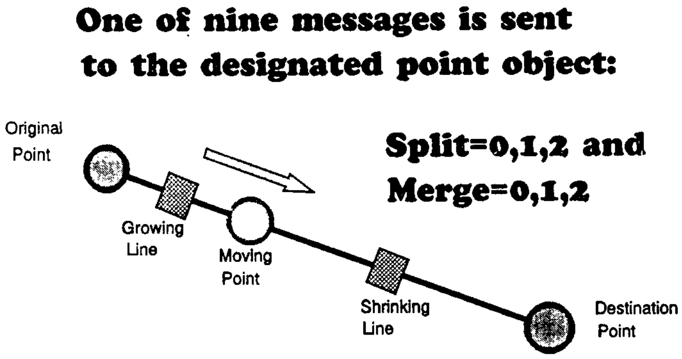


Figure 3. The nine basic Voronoi construction commands.

3.2 Change

Change is treated as the addition or deletion of map objects over time, or the movement of one or more points over time, as with robot simulation or fluid modelling. Lines are created as the locus of a moving point, and point movement in the Voronoi spatial model is treated as a sequence of "topological events", or triangle-pair switches. Roos (1990) worked with simultaneously moving points, and the same approach may be used, one point at a time, together with the nine split/merge options, to add or delete points or line segments. Polygons, when defined, are specified as the set of Voronoi tiles of points and half-lines that are entirely enclosed by line boundaries.

3.3 Basic spatial queries

The basic query is the local "proximal" query, returning the nearest object any time the cursor is moved. The "neighbour" query returns the set of neighbouring objects to an existing object specified with the proximal query. Another operation inserts the query point into the map, creating its own Voronoi region. This is used primarily for various types of interpolation. A fourth local query is "scan", where the topological network is searched radially outwards from some starting location, until some terminating condition (e.g. moving out of a search box, or perhaps reaching a linear boundary). This last is used for polygon labelling, as a polygon is defined as a set of Voronoi zones bounded by line segments. The fifth local query is "trace", where all Voronoi zones crossed by an imaginary "line segment" are reported.

3.4 Vector GIS emulation

These, along with the construction commands, are used in the emulation of traditional GIS analysis queries involving polygon boundary dissolve, buffer zone generation, and polygon overlay. The first of these is implemented using standard construction commands to delete line segments in a local manner, the second is performed by simple geometric construction within each Voronoi zone of the target object set, and the third by using the "trace" query to draw boundaries of one layer through the second (and then the second layer through the first). The resulting line segments, split and labelled with the appropriate polygon "colours", are reconstructed in a third Voronoi layer. In the current implementation the "source" and "trace-through" layers may be viewed in two windows, while the resulting overlay is being incrementally constructed in the third.

4. SYSTEM ARCHITECTURE

The central Voronoi module is a closed box, accessible as a dynamic linked library (DLL) from the calling system. Messages are passed to it, in the same fashion as to a database engine, and responses are returned. Input messages, as with a database engine, are either updates or (non-invasive) queries. Responses consist of a message, a list of objects (objects found for queries, objects modified for updates), plus a list of coordinate pairs if necessary. Input commands must be addressed to an existing object in an existing layer, and the full information for the command is contained in one or two "cursors" (an object name and/or a coordinate pair). A macro name must be given in the event that a collision is detected during the update.

Within the Voronoi module are three components that may differ between versions. The memory management system preserves the triangles ("topology") and map objects (points, half-lines and coordinate pairs) during map update. This may vary internally depending on the assumptions about system memory, paging, etc. The geometry module contains the small number of arithmetic operations necessary with spatial coordinates (determinant, intersection, etc.) Currently this operates in the Euclidean plane, but versions for the sphere are anticipated, and others are possible. The macro list contains the small number of feasible actions in the case of a collision: line intersection and continuation; create node and snap; cancel and undo the previous command, etc.

In the case of a simple query, a message from the calling system results in a simple response: the list of objects found within the spatial conditions (neighbours, within a box, etc.) For a simple map update the same is true, but where a collision is detected (usually when one line attempts to cross another) the Voronoi module must return with the original command only partially completed in order to report the condition and obtain the correct instruction (macro name). Thus there is a loop within the calling system to permit selection of the appropriate response. In addition, if actions, or graphic display, is desired after each topological event, this must also be permitted by a loop within the calling system.

4.1 Platform selection

This architecture is designed to provide the maximum of flexibility (imposing the fewest possible constraints on the potential applications using this spatial server) while preserving the integrity of the server itself. Ongoing experience indicates that new applications are found every few weeks, and great effort has been taken to impose the fewest possible presuppositions on the basic design. The same considerations applied in the selection of the system platform. We wished to work within the PC environment, as we envisaged decision support system applications equivalent to a "spatial spreadsheet", rather than as a replacement for a large corporate GIS. We wished to preserve the integrity of the spatial server, while permitting ready access to its functionality. This suggested a dynamic linked library (DLL) format, accessible by an interpreter for the development of applications prototypes. This led to the selection of Microsoft's Visual Basic as the application development environment, not least because the same considerations applied to access to existing commercial products - e.g. attribute databases and spreadsheets. These could be purchased as required, and the appropriate modules integrated with the application in the same fashion as the Voronoi DLL, permitting the customer (e.g. graduate student) to concentrate on the application rather than the graphics interface or low-level tools. Preliminary experiments suggest that this approach is appropriate, and simple applications are currently being developed.

5. COMPLEX OPERATIONS

The local nature of the basic construction commands and queries gives great flexibility in designing higher level queries. As the topology is always complete, robot-navigation methods may be used to steer the cursor away from unwanted collisions, or to follow along a pre-existing boundary judged to be sufficiently close to the trajectory specified (thus eliminating unwanted sliver polygons). Robot-navigation problems themselves may be addressed, with or without operator interaction, and with or without ongoing changes in the map data. An outline of a marine GIS using these properties is suggested in Gold and Condal (1994).

Based on the low level messages described previously, using both the Voronoi and database modules, additional layers of application functions may be developed between these modules and the Visual Basic interface. A few simple examples should suffice.

5.1 Buffer zone generation

The previously-mentioned buffer-zone generation is performed by inserting the target object set into a Voronoi layer, and then performing a query to extract a list of the names of all the map objects. For each map object a neighbour query is made, returning a list of all its neighbouring objects (together with a list of the Voronoi cell nodes). From this description of the cell outline, the intersection of the corridor boundary (at the specified distance from the generating object) may be calculated (if it is present at all). For a line object this will be a linear segment, for a point object it will be a circular arc (broken into small line segments). These segments will be inserted into another Voronoi layer, and the operation repeated for the next object in the list. Figure 4 shows a simple buffer zone, and that the point-in-polygon problem may be resolved by searching for the nearest object and identifying its tile attribute.

5.2 Interpolation

For a field-type application, rather than the previous object-based application, the Voronoi layer may be generated as before, composed of points and line segments for which attribute (e.g. elevation) information is available. If an elevation grid is desired, each grid node will be taken in turn and inserted into the Voronoi layer using an "add" command. A list of the neighbours of this cell will found by then sending a "neighbour" query for this inserted point. For each neighbour in the list another neighbour query returns the neighbours and voronoi vertices around this, as in the previous case, and from this its cell area may be calculated. The query point is now removed with a "delete" command, and the original set of neighbours processed again to re-calculate their cell areas. The difference in the areas is the weighting to be assigned to each neighbour's elevation value in order to produce the elevation estimate at the grid node (see Gold 1989 and Gold and Roos 1994 for details). In practice a more efficient algorithm may be developed and implemented as a basic Voronoi query, but the method described here is directly achievable with the basic Voronoi command set.

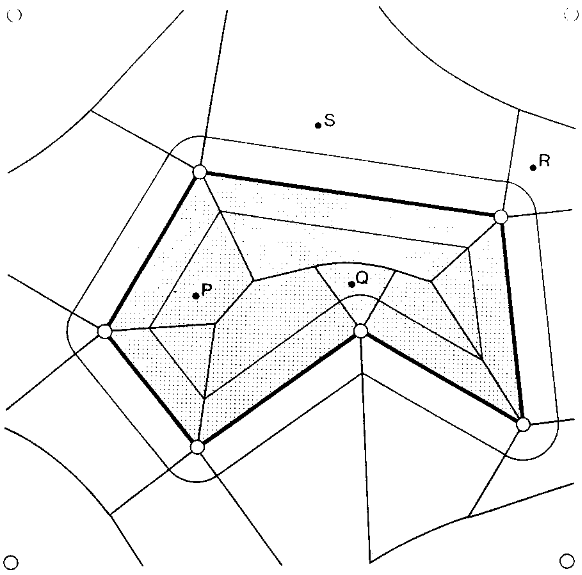


Figure 4. Simple buffer zone about a polygon boundary.

5.3 Polygon overlay

Similar procedures may be developed for polygon overlay, where each line segment from one polygon layer is used in a "trace" command sent to the second polygon layer, which returns with a list of Voronoi cells intersected. This list is used to determine any line intersections and the polygons traversed by the original line segment, which is broken up accordingly and inserted, with the combined set of polygon attributes attached, into the final polygon layer under construction. This process is repeated with each line segment in layer one being traced through layer two, and then layer two segments traced through layer one. The result in the final layer is the combined polygon set. This works for simple overlay cases, but where near-coincidence and related problems occur additional macros are required to allow navigation along existing boundaries within a Voronoi layer. Work is ongoing on this application.

Further complex operations could be described for terrain runoff modelling applications, robot navigation on the basis of both obstacle objects and field-type terrain information, for point cluster analysis and image object recognition, and others that will be reported in detail as they are completed.

6. CONCLUSIONS

The spatial system as currently operational has great flexibility, and may be used as a library of spatial operations, or a "spatial server", to be used by many application shells. Interpolation problems suppose a "field" view of space, with data objects embedded in it. These data objects may be the usual points, but also line segments with specific surface continuity constraints attached - e.g. slope-continuous across contour line input, only surface-continuous across ridge and valley lines, and surface-discontinuous across geological fault lines. Surfaces always honour all data objects and values (Gold and Roos, 1994). The marine application mentioned above includes Voronoi neighbour detection to navigate a boat (a moving query point embedded in the Voronoi map) between obstacles, as well as the same embedded query point to interpolate depth directly from bathymetric observations. Features that are displaced over time may also be handled. Applications under development currently include digital elevation modelling, surface runoff modelling, interactive spatial decision support systems, flow simulation, network analysis, and forest map maintenance.

ACKNOWLEDGEMENTS

This research was made possible by the foundation of an Industrial Research Chair In Geomatics applied to Forestry at Laval University, jointly funded by the Natural Sciences and Engineering Research Council of Canada and the Association de l'Industrie Forestière du Québec.

REFERENCES

Aurenhammer, F., 1991. Voronoi Diagrams - A Survey of a Fundamental Geometric Data Structure. ACM Computing Surveys, v. 23 no. 3, pp. 345-405.

Fortune, S., 1986. A Sweepline Algorithm for Voronoi Diagrams. Proceedings, Second Annual ACM Symposium on Computational Geometry, pp. 313-322.

Gold, C.M., 1989. Chapter 3 - Surface interpolation, spatial adjacency and GIS. IN: Three-Dimensional Applications in Geographic Information Systems (J. Raper, ed.). London: Taylor and Francis.

Gold, C.M., 1992. An Object-based Dynamic Spatial Model, and its Application in the Development of a User-friendly Digitizing System. Proceedings, Fifth International Symposium on Spatial Data Handling, Charleston, August 1992, pp. 495-504.

Gold, C.M. and A.R. Condal, 1994. A spatial data structure integrating GIS and simulation in a marine environment. Marine Geodesy (submitted).

Gold, C.M. and T. Roos, 1994. Surface modelling with guaranteed consistency - an object-based approach. Proceedings, IGIS'94, Ascona, Switzerland, March 1994 (submitted).

Roos, T., 1990. Voronoi Diagrams Over Dynamic Scenes. Proceedings, Second Canadian Conference on Computational Geometry, Ottawa, pp. 209-213.

A TETRAHEDRON-BASED 3D VECTOR DATA MODEL FOR GEOINFORMATION

Morakot Pilouk, Klaus Tempfli
Department of Geoinformatics, International Institute for Aerospace Survey
and Earth Sciences, P.O. Box 6, 350 Boulevard 1945, 7500 AA Enschede,
The Netherlands Tel. +31-53-874358 Fax. +31-53-874335
e-mail: MORAKOT@ITC.NL, TEMPFLI@ITC.NL

Martien Molenaar
Department of Surveying, Photogrammetry and Remote Sensing, Wageningen,
Agricultural University, P.O. Box 339, 6700 AH Wageningen, The Netherlands
Tel. +31-8370-82910 Fax. +31-8370-84643
e-mail: MARTIEN.MOLENAAR@WETENSCH.LMK.WAU.NL

Abstract

In modelling 3D objects, the formal data structure for 3D vector maps is well suited for the representation of buildings and other man-made objects which can be sampled exhaustively because of their regular shapes. When entering the geosciences and environmental management, however, the data model should not only account for topology and thematic properties but also facilitate interpolation. Complex phenomena such as layers of soil are usually sampled sparsely because of costs. Detailed spatial analysis then requires prediction of values in a fine network. Introducing the tetrahedron as a basic structuring element promises efficient interpolation and maintenance of topology. This paper presents the new 3D data model and the corresponding relational 3D data structure. Based on the concept of combinatorial topology, the 3D model is also generalized to a generic n-dimensional model. The second part of the paper describes the necessary tessellation algorithms for building a network of tetrahedrons-- constrained or unconstrained-- from a set of arbitrarily distributed points.

Introduction

The demand for spatial information is increasing, and many applications require full 3D information while most of the existing geo-information systems are still confined to 2D. Tools for 3D data manipulation can be found in computer graphics and CAD systems. Such systems, however, cannot readily be applied to handle geo-information. CAD systems aim at the design and visualisation of spatial objects and deal with relatively small data sets, while geo-information systems focus on the relationships among spatial objects and their thematic properties. GISs deal with inventory and analysis of geographic phenomena and large and complex data sets. Objects in a GIS are often of other dimensions and characteristics than in CAD, and even other coordinate systems have to be used in describing geometry when considering coverages of large areas. Therefore, developments that are more suitable to 3D geo-information are needed.

The development process starts with modelling, *ie*, abstracting reality. The formal data structure for 3D vector maps (3D FDS; Molenaar, 1992) is an attempt in 3D modelling suited for geo-information in many aspects. The 3D FDS, however, was not designed to also handle interpolation processes efficiently. This is a requirement of only some applications, specifically in geo-science and environmental management. There, phenomena can often be sampled only sparsely and more detailed information has to be derived computationally. A data model is desirable that can preserve the original spatial information and also accommodate the derived information.

The need for interpolation is well-known from digital terrain relief modelling. Interpolation in DTM can easily be accomplished if data are structured as a triangular irregular network (TIN). A TIN, with its elements nodes, arcs, and triangles, can also serve geometric modelling of geographic phenomena such as rivers, roads, land cover, etc., which are objects of many GISs. We can follow the same reasoning that leads to the unified data structure of terrain relief and terrain features -- thus, the "integration of DTM and GIS" (Pilouk and Tempfli, 1993) – in making the transition from surface to volume models. By extending the 3D FDS to include the geometric primitives, triangle and tetrahedron, we aim at a data model that also allows efficient 3D interpolation. Recently tetrahedral approaches to 3D modelling were also suggested by, *eg*, Raper (1993), Midtbø (1993), Tsai and Vonderohe (1993), Kraak (1992).

This paper first reviews the 3D FDS and then presents the tetrahedron-based model, the single-theme and the multi-theme concepts. The new model can be described within the mathematical theory of combinatorial topology, which offers the opportunity to generalize it to n dimensions. In order to indicate the implementation of this conceptual model, it is mapped to a relational database structure. The availability of a fast and robust algorithm to "tetrahedronize" a data set is important for the operationalization. The second part of the paper describes how this can be accomplished using raster processing techniques.

Formal data structure for 3D vector maps

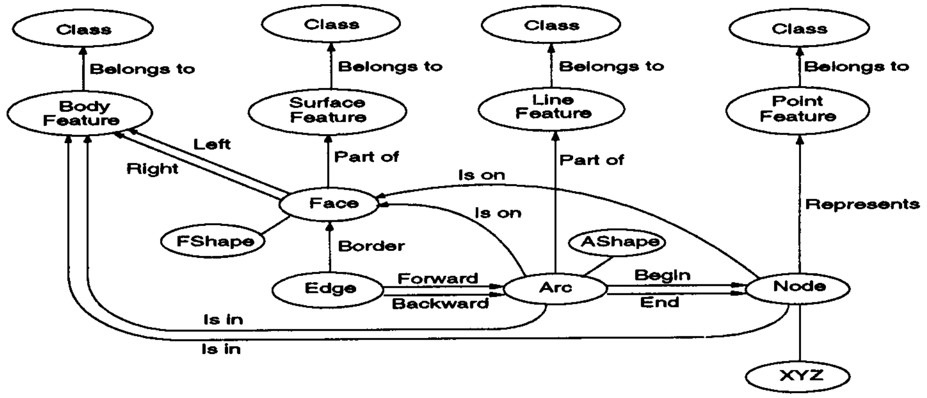


Figure 1 : Formal data structure for 3D vector map

the conceptual design of the 3D FDS, as shown in figure 1, is based on the decomposition of an object of the real world into identifier, geometry and theme. An object identifier maintains the link between the geometric and the thematic descriptions. Accordingly, the object components are grouped into three levels, namely the geometric, feature and class levels. On the geometric level we distinguish the geometric primitives *nodes*, *arcs*, and *faces* that are respectively of 0, 1 and 2 dimensions. *Edges* are additional geometric primitives that provide the link between arcs and faces thus allow the unique reference to *left* and *right* body features. Unless faces are constrained to be planar surfaces, the primitive *fshape* provides a mathematical or numerical description of a face. On the feature level, each feature is represented by its identifier. We distinguish four types of feature: *point*, *line*, *surface* and *body*. A point feature consists of a node, a line feature consists of one or more arcs, a surface feature consists of one or more faces, and a body feature is bounded by faces. Within the concept of a single-valued vector map (see Molenaar, 1989), each feature is then related to one specific thematic class. Bric et al (1994) have shown the applicability of this model for problems where the involved spatial objects are regularly shaped, and when geometry, topology and thematic attributes are known at the time of data acquisition.

Spatial interpolation

There are many other 3D problems in which the geometry and topology of the objects of interest are not known at the time of data acquisition. This applies when objects are not visible or not directly perceivable, *eg*, sulphur in the air, underground soil strata, concentration of chemical contamination in water, etc. In such cases measurements cannot be taken of the object itself (*eg*, the hull of the sulphur cloud), but only at fixed spatial positions. Based on the sample taken and assumptions about the behaviour of the phenomenon in question (*eg*, how sulphur spreads in the air), the shape of the object (the sulphur cloud) can be determined computationally by interpolation. Thereafter topology can be built and the thematic properties can be assigned.

Interpolation can be done in many different ways. The fidelity of the result will depend on the mathematical model used and how reference data are selected (see Tempfli and Makarovic, 1978). Unless all data are used simultaneously, the selection of reference data for a simple interpolation should be based on proximity. A well-known approach to establishing a proximal order of a 2D point set is the Delaunay triangulation. Once a data set has been triangulated, a variety of interpolation methods can be applied, ranging from very simple to highly sophisticated ones. When interpolating a surface, the problem can be posed as estimating a single-valued function defined on a 2D reference space, *ie*, $Z = f(x, y)$. Z, *eg*, can be the elevation of the ground surface. A simple, yet adequate method for several applications is to compute Z at a position (x, y) as the weighted average of Z at the nodes of the triangle (x, y) belong to.

When dealing with solid objects, the interpolation problem becomes trivariate, *ie*, we must estimate $p = f(x, y, z)$. Accordingly we have to establish a 3D proximal order prior to interpolation. This can be attained by generalizing the Delaunay triangulation to construct a network of tetrahedrons such that the circumsphere of any tetrahedron does not contain any other data point than the four nodes of the tetrahedron. Once proximity is defined by a tetrahedral network (TEN), several bivariate interpolation methods can readily be generalized to operate on trivariate data, *eg*, the weighted average.

Delaunay triangulation is the geometrical dual of the Voronoi tessellation. The proximity properties of the Voronoi (or Dirichlet or Thiessen) polygons have frequently been discussed in literature, so further elaboration will be omitted here. The approach to 3D Voronoi tessellation and the derivation of TEN from the Voronoi polyhedrons will be described later in this paper.

The tetrahedron-based data model

The data model shown in figure 2 is obtained from figure 1 by decomposing bodies into tetrahedrons and, accordingly, surfaces into triangles. By introducing these new geometric primitives the relationships become more transparent (the face, edge and shape primitives become redundant) and interpolation becomes much easier. To ensure that all original object descriptions are maintained, known point, line and surface features are constrained into vertices (nodes), edges of triangles (arcs), and planar triangles of tetrahedrons. This model can also be seen as the 3D generalization of the TIN-based integrated surface data model (see Pilouk and Tempfli, 1994).

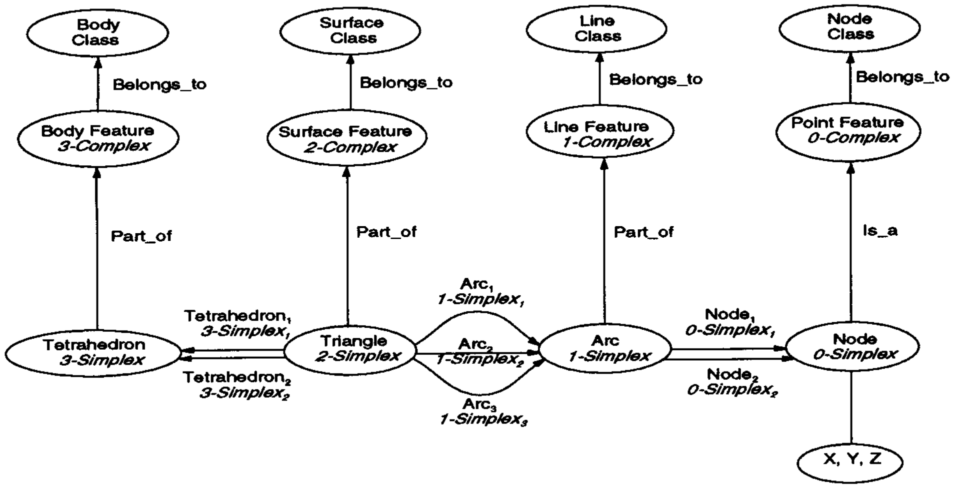


Figure 2 : A tetrahedron-based data model

The data model shown in figure 2 can be explained either descriptively or mathematically. Following the FDS approach, we can describe the relationships among the data types in the form of rules and conventions:

(1) An instance of the node data type is composed of x, y, and z coordinate types. It may be a part of an instance of a point feature type.
(2) An arc data type is defined as a straight line; it is therefore composed of only two instances of the type node, one on each end. It may be defined as a part of an instance of a line feature type.
(3) A triangle data type is composed of three arcs. It is shared by two tetrahedrons, one on each side of its plane (called 1st- and 2nd-tetrahedron, respectively). It may be a part of a surface feature.
(4) A tetrahedron is a part of a body feature.

We should also note that the tetrahedron data type does not carry any geometric description (*ie*, triangular faces, edges, vertices) since its components can always be found from the geometric links with the triangle data type (being either the first or the second tetrahedron of a triangle). Further discussion about the links among the data types is given below in the section on the relational database structure.

The mathematical description of this data model follows the theory of combinatorial topology (see Egenhofer et al, 1989). The approach is based on the classification of spatial objects according to their spatial dimension. We can define a minimal object, called simplex, for each dimension. For example, a node is a 0-simplex, an arc (a straight line consisting of two nodes) is a 1-simplex, a triangle is a 2-simplex and a tetrahedron is a 3-simplex. Spatial position is defined by linking nodes to coordinates. Based on the notion of minimal objects and that a minimal object in a higher dimension is composed of many minimal objects from lower dimensions, the following definitions can be given:

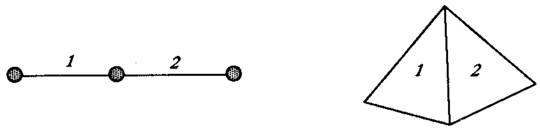
Definition #0: The dimension is defined by the number of components of the coordinate tuple.

For example, nodes are defined by coordinate pairs in 2-dimensional space, in 3-dimensional space by (x, y, z), in n-dimensional space by an n-tuple.

Definition #1: Any simplex of dimension n, called n-simplex, is bounded by (n+1) geometrically independent simplices of dimension (n-1) (Egenhofer et al, 1989) and n+1 simplices of dimension 0.

For example, a tetrahedron (3-simplex) is bounded by four triangles (2-simplices) and four nodes (0-simplices), a triangle (2-simplex) is bounded by three arcs (edges of a triangle, 1-simplices) and three nodes, an arc (1-simplex) is bounded by two nodes. Arcs, *eg*, are geometrically independent if they are not parallel and none of them is of length zero.

Definition #2: Confining analysis to an n-dimensional space, an (n-1)-simplex can be shared by at most only two n-simplices.



R^1
a 0-simplex is shared by 2 1-simplices

R^2
a 1-simplex is shared by 2 2-simplices

R^3
a 2-simplex is shared by 2 3-simplices

R^4
a 3-simplex is shared by 2 4-simplices

Figure 3 : Examples of n-simplex being shared by two of (n+1)-simplices

For example, in 1-dimensional space ($x \neq 0$), a node can be shared by at most two straight-line segments (whereas in two or higher dimensional space, a node can be shared by an infinite number of arcs); in a 2-dimensional space, an arc can be shared by only two triangles; in 3-dimensional space, a triangle can be shared by only two tetrahedrons. Similarly, in a 4-dimensional space, a tetrahedron can be shared by only two 4-simplices (see figure 3 for graphic illustration).

Note that the above definitions hold for simplices only. They do not hold for complexes. Complexes are set of simplices of the same dimension.

Generalized data model for n-dimension

Given the above three definitions, a generic n-dimensional data model can be derived following the logic we observed when extending our model from 2D to 3D. Figure 4 illustrates the *n*D data model. The generic data model can be illustrated very elegantly and it has the advantage that objects of dimension higher than three need not be given names.

Single-theme and multi-theme

Figure 4 shows the *n*D data model for the single-theme concept. Figure 5 shows the corresponding multi-theme data model. The characteristic of a single-theme data model is that an instance of a feature type belongs to only one thematic class, and an instance of a geometric type (*ie*, node, arc, triangle, tetrahedron) can be defined as a part of only one instance of a feature type. For a multi-theme data model, an instance of a feature type still belongs to only one thematic class, but an instance of a geometric type can be defined as a part of one or more instances of a feature type (for detailed description of the multi-theme concept, see Bouloucos et al, 1993).

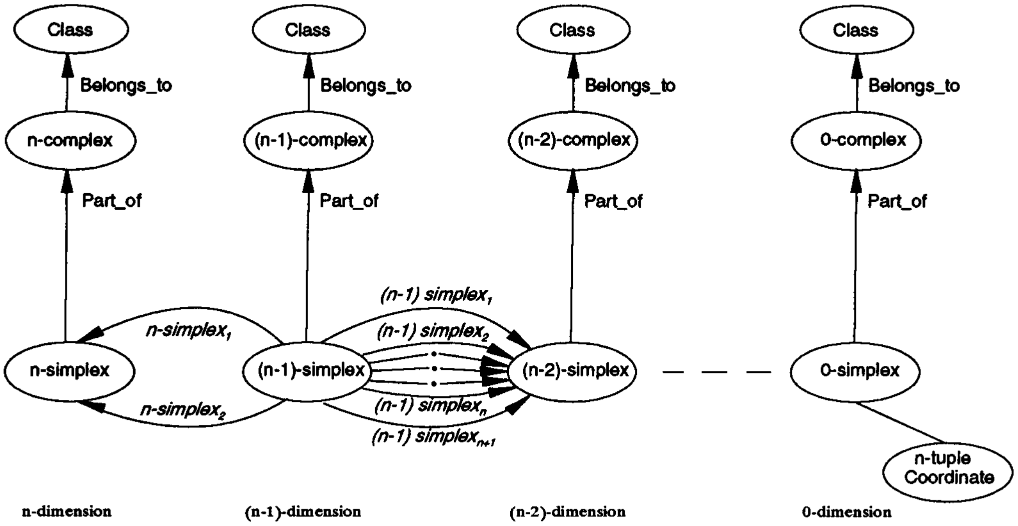


***Figure 4** : A generalized n-dimensional data model for single-theme*

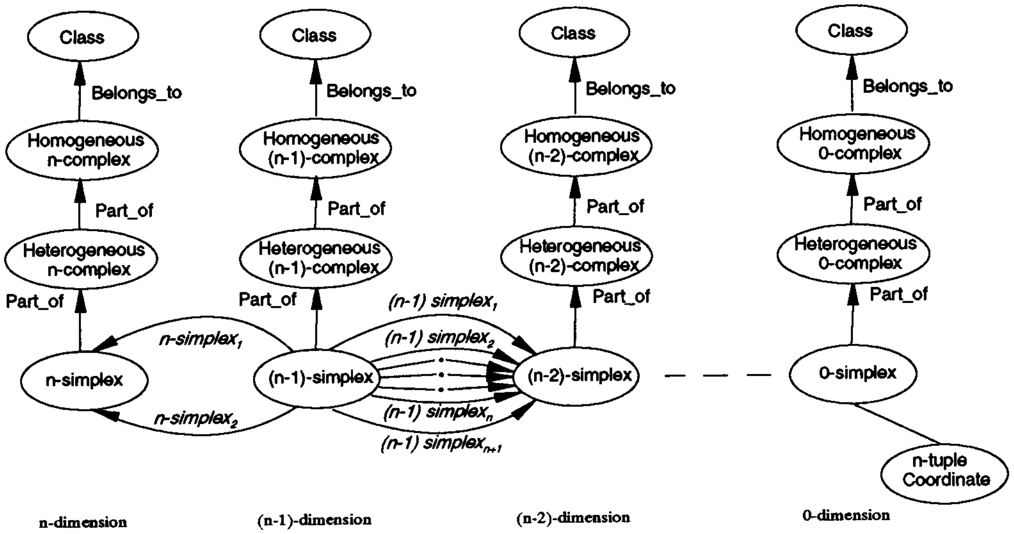


Figure 5 : Generic multi-theme data model for n dimension

The multi-theme data model can be seen as an extension of the single-theme data model, as it accepts objects that share the same spatial region. This means two or more objects can have overlapping parts (*ie*, body, surface, line, point). A typical example is that layers of soil and volume of ground water are sharing the same spatial region. Within the multi-theme concept, two types of complexes must be distinguished. A homogeneous complex (feature) is a set of contiguous simplices of the same dimension and all relating to only one class. A heterogeneous complex (overlapping part) is a set of contiguous simplices of the same dimension that relate to more than one class. A heterogeneous complex is part of two or more homogeneous complexes. By introducing homogeneous and heterogeneous complexes, we can solve the problem of "many-to-many" relationships between geometric primitives and features.

Relational Database Structure

In general, the data models of figures 1, 2, 4 and 5 can be mapped to different logical models of a database (*ie*, hierarchical, network or relational). In this paper, we outline the mapping of the data model of figure 2 into a relational database structure. The following 10 relations (tables) are obtained from the normalization process that needs to be applied to avoid redundancy and ensure database integrity during updating processes. The approach, using dependency statements and diagram (Smith, 1985; Pilouk and Tempfli, 1993), is applied since it offers a rigorous and simpler process than non-lost decomposition (Date, 1986).

R1: Node (NodeNr, x, y, z)
R2: Arc (ArcNr, node1, node2)
R3: Triangle (TriNr, tet1, tet2, arc1, arc2, arc3)
R4: Tetra (TetNr, bid)
R5: TriSurf (TriNr, sid)
R6: ArcLine (ArcNr, lid)
R7: Point (Pid, pclass, pnode)
R8: Line (Lid, lclass)
R9: Surface (Sid, sclass)
R10: Body (Bid, bclass)

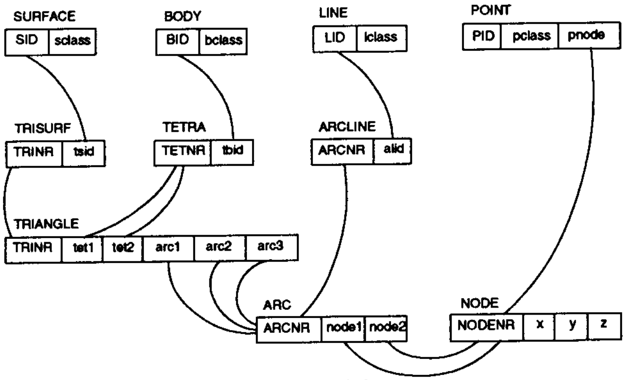


Figure 6 : Relational database model

R1, R2 and R3 can be regarded as geometry tables. R4, R5 and R6 are geometry-feature tables. R7 is a geometry-feature-class table. R8, R9 and R10 are class tables. Concerning the n-dimensional model, it is possible to predict the number of tables necessary for the data structure. There are *n* geometric tables, *n* geometry-feature tables, *1* geometry-feature-class table (*ie*, for the link between node and point feature and class), and *n* class tables. Note that table R4 maintains no other information than just a tetrahedron number (*TetNr*) and an identifier of the body feature (*bid*) it belongs to. This means that we can search for the geometric components of a tetrahedron of interest only via the R3 (Triangle) table, by matching the attribute value of the tetrahedron (*TetNr*) with either attribute value of *tet1* or *tet2*. Once the match is found, the next step is to get each of the three attribute values of *arc1*, *arc2* and *arc3* of the R3 table as a key to search for the match with the *ArcNr* in the R2 (Arc) table. If the match is found we must get the attribute value of *node1* and *node2* and use each of them to search for the match with the *NodeNr* in the R1 (Node) table to get the coordinates x, y, and z. In this way, we can use this database for 3D interpolation and for responding to a wide range of queries.

3D Voronoi tessellation

Tsai and Vonderohe (1991) and Midtbø (1993) suggested a vector approach to tetrahedronize a set of points. A difficulty of the vector approach is that some data sorting and spatial indexing is already required prior to building Delaunay tetrahedrons. A TEN, which complies with the Delaunay condition of "empty circumspheres", can be derived from the Voronoi polyhedral tessellation in the same way as deriving Delaunay triangles from Voronoi polygons. This offers the possibility to design an algorithm based on a raster approach that is simple and easy to implement. We can generalize the approach presented by Pilouk and Tempfli (1992), and Tang (1992) for deriving a TIN, using the distance transformation (see Borgefors, 1984, 1986). Given a set of points defined by (x, y, z) coordinates, the first step is a vector-to-raster conversion, choosing a voxel size small enough so that the two closest points are not mapped to the same voxel. In order to determine proximal polyhedrons for the set of points, two 3D raster images are generated simultaneously in the next step, the distance image and the Voronoi tessellation. The distance image shows shortest distances, *ie*, for every voxel of the entire image the distance to the nearest data voxel (given point). It is used to build up, step-by-step, the tessellated image that shows the polyhedron of influence for every data voxel. Any voxel inside a particular polyhedron is closer to the data voxel of this polyhedron than to any other data voxel.

The 3D distance transformation is based on computing the "Chamfer distances", which are approximations of Euclidean distances. It is carried out by scanning the original raster image in two passes by a 3x3x3 mask. To this end the mask is split into two parts (see figure 7). The asymmetric mask is called "Chamfer 3-4-5", indicating the way the distance approximation is made. Three approximate values for all the neighbours of a voxel under transformation are distinguished, 3 for the ones that have the connection to the central voxel (which gets the value zero) by a face, 4 for those connecting by edges, and 5 for those connecting by vertices.

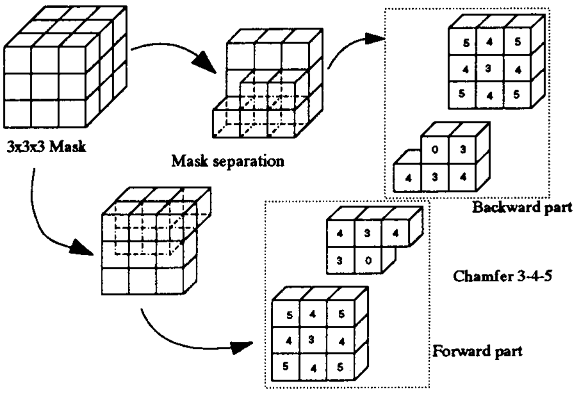
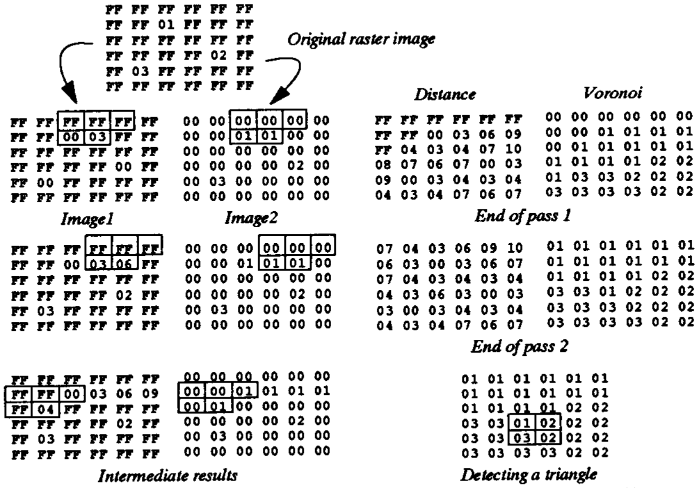


***Figure 7** : 3x3x3 Mask for 3D distance transformation and Voronoi tessellation*

The original raster image has voxel values equal to the highest integer (*ie*, 65536, respectively FF in hexadecimal, for unsigned 2-byte integer numbers) except at data voxels, which are assigned the original point identifier. In the forward scan the mask is moved systematically through the entire voxel array. For each location of the mask, the value of each element of the mask is added to the corresponding element of the input voxel array. Any value resulting from the additions is a candidate and the smallest value is selected for the best distance approximation at that location. The backward scan follows the reverse route using the second part of the Chamfer mask.



```
FF FF FF FF FF FF
FF FF 01 FF FF FF
FF FF FF FF FF FF
FF FF FF FF 02 FF
FF 03 FF FF FF FF
FF FF FF FF FF FF
```

```
FF FF FF FF FF FF    00 00 00 00 00 00    FF FF FF FF FF FF    00 00 00 00 00 00
FF FF 00 03 FF FF    00 00 01 01 00 00    FF FF 00 03 06 09    00 00 01 01 01 01
FF FF FF FF FF FF    00 00 00 00 00 00    FF 04 03 04 07 10    00 01 01 01 01 01
FF FF FF FF 00 FF    00 00 00 00 02 00    08 07 06 07 00 03    01 01 01 01 02 02
FF 00 FF FF FF FF    00 03 00 00 00 00    09 00 03 04 03 04    01 03 03 02 02 02
FF FF FF FF FF FF    00 00 00 00 00 00    04 03 04 07 06 07    03 03 03 03 02 02
```

```
FF FF FF FF FF FF    00 00 00 00 00 00    07 04 03 06 09 10    01 01 01 01 01 01
FF FF 00 03 06 FF    00 00 01 01 01 00    06 03 00 03 06 07    01 01 01 01 01 01
FF FF FF FF FF FF    00 00 00 00 00 00    07 04 03 04 03 04    01 01 01 01 02 02
FF FF FF FF 02 FF    00 00 00 00 02 00    04 03 06 03 00 03    03 03 01 02 02 02
FF 03 FF FF FF FF    00 03 00 00 00 00    03 00 03 04 03 04    03 03 03 02 02 02
FF FF FF FF FF FF    00 00 00 00 00 00    04 03 04 07 06 07    03 03 03 03 02 02
```

```
FF FF FF FF FF FF    00 00 00 00 00 00    01 01 01 01 01 01
FF FF 00 03 06 09    00 00 01 01 01 01    01 01 01 01 01 01
FF 04 FF FF FF FF    00 01 00 00 00 00    01 01 01 01 02 02
FF FF FF FF 02 FF    00 00 00 00 02 00    03 03 01 02 02 02
FF 03 FF FF FF FF    00 03 00 00 00 00    03 03 03 02 02 02
FF FF FF FF FF FF    00 00 00 00 00 00    03 03 03 03 02 02
```

***Figure 8** : Parallel processing of distance transformation and Voronoi tessellation; an example for a 2D case.*

As the mask is moved (in the forward and backward scan) the second image is build up voxel by voxel by keeping track of the data voxel that determines the shortest distance. The corresponding voxel in the second image is assigned the identifier of the closest data voxel; see figure 8 as an illustration for the 2D case.

Formation of tetrahedrons

By scanning the tessellated image once by a 2x2x2 conditional mask, the TEN is obtained in the designated data structure. The network will be an unconstrained one because no a priori restriction was applied in the Voronoi tessellation. There are six primary conditions attached to the 2x2x2 mask (see figure 9) in order to form a unique set of tetrahedrons (connecting natural neighbour data voxels and no intersecting tetrahedrons). Note that the numbers encompassed by circles in figure 9 correspond to the following conditions (1) to (6). The numbering system of the mask is shown by the numbers encompassed by cubes. The lines, which are drawn between the elements of the mask, show the possible edges of tetrahedron(s) that can be formed by this mask when scanning the Voronoi tessellated image.

(1) $1 \neq 3 \neq 4 \neq 5$
(2) $1 \neq 2 \neq 4 \neq 5$
(3) $3 \neq 4 \neq 5 \neq 7$
(4) $2 \neq 4 \neq 5 \neq 6$
(5) $4 \neq 5 \neq 6 \neq 8$
(6) $4 \neq 5 \neq 7 \neq 8$

The mask looks for the boundaries of the polyhedrons. When the mask discovers a position at which not all voxel values are the same, the neighbours are checked as candidates to form the edge of a tetrahedron. In general, the above six conditions are based on the consideration that there should be at least four different voxel values at the position of the mask to be sufficient to form a tetrahedron. These four different voxel values represent four adjacent Voronoi polyhedrons from which a tetrahedron can be derived. It can happen, however, that there are more than four different voxel values at that time, which implies that more than four adjacent Voronoi polyhedrons are detected. This allows formation of tetrahedrons in several ways, which results in confused topology (comparable to the situation in 2D where four points close to a square give rise to two possibilities of forming two triangles). The following conditions are therefore added to prevent overlapping or intersecting tetrahedrons, thus permitting only the necessary number of tetrahedrons to be formed. These conditions also prevent the formation of tetrahedrons when a sliver Voronoi polyhedron (mostly due to the imperfection of the distance approximation) is present. Conditions (a) to (i) correspond to the letters which are placed on the edges of tetrahedrons in figure 9.

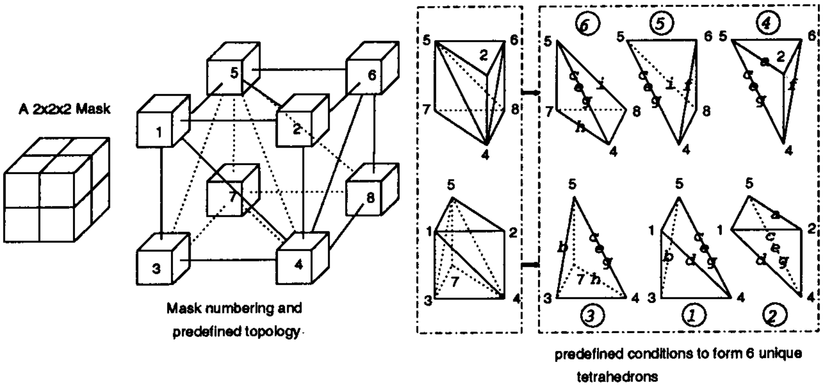


Figure 9 : A 2x2x2 conditional mask used to form tetrahedral network

(a) $1 \neq 6$, *(b)* $1 \neq 7$, *(c)* $1 \neq 8$,
(d) $2 \neq 3$, *(e)* $2 \neq 7$, *(f)* $2 \neq 8$,
(g) $3 \neq 6$, *(h)* $3 \neq 8$, *(i)* $6 \neq 7$.

Apart from the above conditions, there are yet another three to be added. These three conditions are needed to prevent the formation of a tetrahedron because of raster peculiarities. Similar to the broken appearance of an inclined line in a 2D raster image, in 3D an inclined plane appears as a staircase. This causes a problem when four points are situated on an inclined plane (relative to the 2x2x2 mask) and by chance on the circumference of a circle. Then four adjacent Voronoi polyhedrons would be detected. Without the three conditions given below, a flat tetrahedron would be formed. This problem does not occur if there are completely horizontal or vertical planes (relative to the mask), since the previous conditions take care of such a constellation:

(j) not $(1 \neq 5$ *and* $(1 = 4)$ *and* $(5 = 8))$,
(k) not $(3 \neq 4$ *and* $(3 = 5)$ *and* $(4 = 6))$,
(l) not $(2 \neq 4$ *and* $(2 = 5)$ *and* $(4 = 7))$.

Combining the first, the second and the third sets of conditions leads to the following algorithm:

```
if      (1) and (b) and (c) and (d) and (e) and (g) and (l)
 or     (2) and (a) and (c) and (d) and (e) and (g) and (k)
 or     (3) and (b) and (c) and (e) and (g) and (h) and (j)
 or     (4) and (a) and (c) and (e) and (f) and (g) and (j)
 or     (5) and (c) and (e) and (f) and (g) and (i) and (l)
 or     (6) and (c) and (e) and (g) and (h) and (i) and (k)
then
        increase number of tetrahedrons
        form a tetrahedron.
```

Since there are several alternatives in designing this mask (*eg*, a cube can be decomposed into five or six tetrahedrons), the mask in figure 9 takes into account the compatibility with 2D triangulation. This provides an easy way to combine a TIN with TEN without any conflict. The design of the 2x2x2 mask is based on the principle that a cube can be cut by three different planes, each plane passing through two diagonally opposite edges of the cube, considering a pair of edges for every coordinate axis. The three planes intersect each other along a diagonal of the cube and divide the cube into six tetrahedrons. After finishing the TEN generation, Euler's equality (which we generalized for the n-dimensional case) can be used to check the internal consistency of the network.

$$Nodes + Triangles = Arcs + Tetrahedrons + 1$$

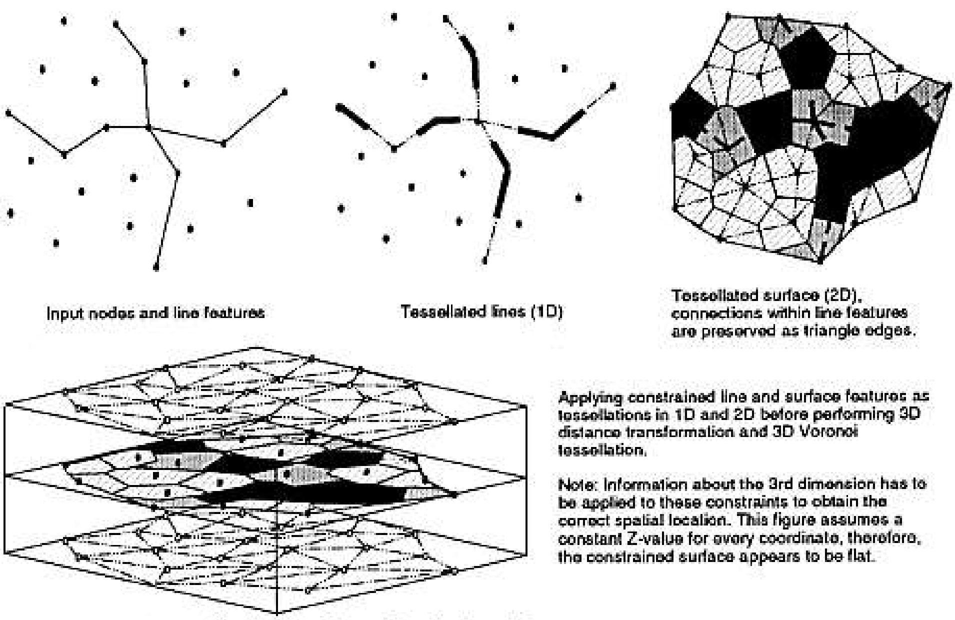
For n-dimension, the equality becomes:

$$0_{simplices} + 2_{simplices} + \dots + k_{simplices} = 1_{simplices} + 3_{simplices} + \dots + l_{simplices} + 1$$

where: k is an even number; $k = (n+2)*2$; $\{k \le n\}$
l is an odd number; $l = ((n+1)+2)*2-1$; $\{l \le n\}$

If Euler's condition is not met, it indicates that there are either free points or overlapping tetrahedrons present in the TEN.

Constrained tetrahedronization



Applying constrained line and surface features as tessellations in 1D and 2D before performing 3D distance transformation and 3D Voronoi tessellation.

Note: Information about the 3rd dimension has to be applied to these constraints to obtain the correct spatial location. This figure assumes a constant Z-value for every coordinate, therefore, the constrained surface appears to be flat.

Figure 10 : Voronoi tessellation in dimension 0, 1 and 2

In order to maintain features that are already known from data acquisition (eg, the contact surface between two layers of soil, a ground water surface, the ground surface, the drainage network, etc.), constrained tetrahedronization has to be applied. We can generalize again the approach described by Tang (1992), and Pilouk and Tempfli (1994) to 3D. The additional constraint required herein is to maintain a triangular facet. Such a triangular facet can originate from a triangulated surface (which has been classified and thus contains information about an area feature) and should not be altered by the tetrahedronization. The proposed solution is to apply a "Voronoi tessellation at lower dimensions" beforehand. By applying the "half-line" concept to preserve the straight line connection between two nodes, we can say that the corresponding arc was previously tessellated in 1D. The influence zone of either "begin" or "end" node covers the distances from that node to the middle of the line or any other line that emanates from that node (see figure 10). To put the constrained tessellation into practice, every single raster element of the influence zone has to inherit the pixel value from its kernel (node). This information has to be added to the original raster image to obtain the tessellation in a higher dimension. This concept holds for n dimensions. The practical steps in 3D are: first tessellate in 1D to obtain constrained arcs. Next, supply the constrained arcs as input for the 2D distance transformation and tessellate the 2D data to obtain constrained triangles. Then merge this information with the input data for the 3D distance transformation. For the 3D distance transformation, the 3D coordinates (x, y, z) have to be used when adding the tessellation in the lower dimension to the 3D raster image. The effect of marking the zones of influence of the salient nodes and arcs is that they behave under distance transformation like individual data voxels (every voxel of an influence zone will get the distance value zero). This way barriers are introduced preventing Voronoi regions being generated across constrained arcs and triangles. The adjacent primitives of a constrained feature create their Voronoi polyhedrons such that they are also adjacent. This ensures that their adjacency will be detected by the 2x2x2 mask during the formation of tetrahedrons (see figure 11).

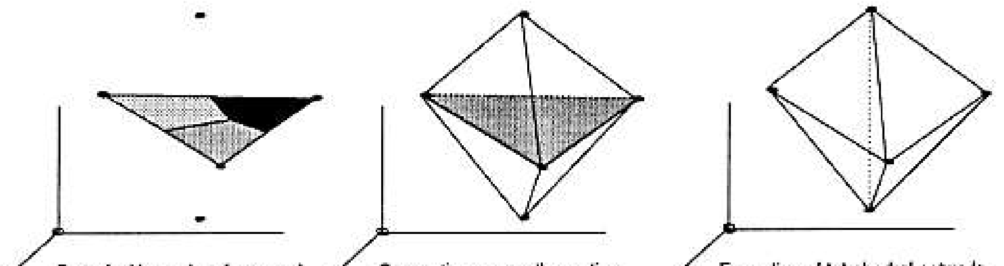
Concluding remarks

As can be seen from figure 2, the TEN-based data model supports normal GIS operations such as queries and provides possibilities to derive additional information by interpolation. It also makes relationships very transparent, thus allowing generalization to n dimensions. A disadvantage is the unnecessarily larger amount of data for well defined, regularly shaped objects and consequently longer response times for queries. The formation of TEN using a raster approach has several advantages over the vector approach, among them:

- the algorithm extensively exploits Boolean operations, complicated algebraic calculations are not needed, thus execution speed is high;
- the constraints can be added objectively.

Some disadvantages are:

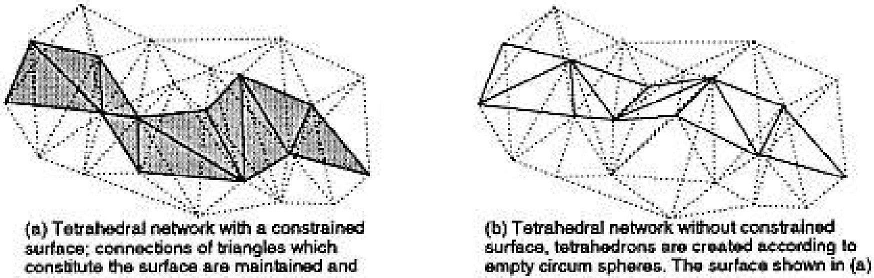
- a lot of storage space is required for the intermediate result, an aspect inherent to all operations based on voxels;
- the performance is resolution-dependant; the user needs to define the resolution based on the required accuracy.



Part of a Voronoi surface used as constraint during formation of the tetrahedral network

Connection among the vertices of a triangle, being part of the constrained surface, is preserved

Formation of tetrahedral network without constraint



(a) Tetrahedral network with a constrained surface; connections of triangles which constitute the surface are maintained and used as faces of tetrahedrons.

(b) Tetrahedral network without constrained surface, tetrahedrons are created according to empty circum spheres. The surface shown in (a) is lost.

Figure 11 : Adding constraint surface as Voronoi regions

The algorithms to create unconstrained TEN have already been implemented using Pascal programming language. The set of programs of 3D raster conversion, creating 3D Voronoi polyhedrons, generating TEN, and visualizing the results have been tested. They proved the applicability of the algorithms. Future works are directed to the implementation of generating TEN with constraints and TEN-based databases.

References

Borgefors, G., 1984. Distance Transformations in Arbitrary Dimensions, Computer Vision, Graphics, and Image Processing, 27, pp. 321-345.

Borgefors, G., 1986. Distance Transformations in Digital Images, Computer Vision, Graphics, and Image Processing, 34, pp. 344-371.

Bouloucos, T., Ayugi, S.W.O., and Kufoniyi, O., 1993. Data Structure for Multi-valued Vector Maps, Proc. EGIS'93, Genoa, Italy, pp. 237-245.

Bric, V., Pilouk, M., and Tempfli, K., 1994. Towards 3D-GIS: Experimenting with a Vector Data Structure, ISPRS Vol. XXX, Part 4, Athens, Georgia, USA, pp. 634-640.

Date, C.J., 1986. An Introduction to Database Systems, Vol. 1, Addison-Wesley, Reading, Mass.

Egenhofer, M.J., Frank, A.U., and Jackson, J.P., 1989. A Topological Data Model for Spatial Databases, NCGIA technical report No. 104.

Kraak, M-J., 1992. Working with triangulation-based spatial data in 3D space, ITC Journal 1992-1, pp. 20-33.

Molenaar, M., 1989. Single Valued Vector Maps - a Concept in GIS, Geo-Informations-Systeme, Vol. 2, No. 1, pp. 18-27.

Molenaar, M., 1992. A topology for 3D vector maps, ITC Journal, 1992-1, pp. 25-33

Pilouk, M., and Tempfli, K., 1992. A Digital Image Processing Approach to Creating DTMs from Digitized Contours, ISPRS Vol. XXIX, Part B4, Comm. IV, Washington, D.C., U.S.A., pp. 956-961.

Pilouk, M., and Tempfli, K., 1993. An Integrated DTM-GIS Data Structure: A Relational Approach, AUTOCARTO 11, Minneapolis, Minnesota, USA, pp. 278-287.

Pilouk, M., and Tempfli, K., 1994. Integrating DTM and GIS Using a Relational Data Structure, GIS'94, Vol. 1, Vancouver, Canada, pp. 163-169.

Raper, J., 1993. Three Dimensional GIS for the 1990, Seminar on 3 Dimensional GIS Recent Developments, ITC, Delft, The Netherlands, pp. 4-5.

Smith, H.C., 1985. Data base design: Composing fully normalized tables from a rigorous dependency diagram, Communication of ACM, Vol. 28, No. 8, pp. 826-838.

Tang, L., 1992. Raster Algorithms for Surface Modelling, ISPRS, Vol. XXIX, Part B3, Comm. III, Washington, D.C., U.S.A., pp. 566-573.

Tempfli, K., and Makarovic, B., 1978. Transfer functions of interpolation methods, ITC Journal 1978-1, pp. 50-78.

Midtbø, T., 1993. Incremental Delaunay Tetrahedrization for Adaptive Data Modelling, EGIS'93, Genoa, Italy, pp. 227-236.

Tsai, V.J.d., and Vonderohe, A.P., 1991. A Generalized Algorithm for the Construction of Delaunay Triangulations in Euclidean n-Space, Proc., GIS/LIS '91 Annual Conference, Atlanta, GA, Vol. 2, pp. 562-571.

Tsai, V.J.d., and Vonderohe, A.P., 1993. Delaunay Tetrahedral Data Modelling for 3-D GIS Applications, Proc., GIS/LIS '93 Conference, Minneapolis, Minnesota, Vol. 2, pp. 671-680.

Modelling of Fuzzy Data

AGGREGATION AND DISAGGREGATION OF FUZZY POLYGONS FOR SPATIAL-TEMPORAL MODELING

Geoffrey Edwards
Chaire industrielle en géomatique appliquée à la foresterie
Centre de recherche en géomatique
Pavillon Casault, Sainte-Foy
Québec, G1K 7P4
e-mail: edwardsg@vm1.ulaval.ca

ABSTRACT. Procedures are presented for constructing fuzzy overlays of maps of different dates. Both raster and vector methods are described and compared. The vector method is used on a sequence of three forest maps generated at intervals of ten years. The fuzzy overlay procedure reduces the number of such transition polygons by nearly a factor of ten compared to standard overlay procedures. The fraction of overlaid polygons which are coherent from one date to another are also examined. It is found that boundaries are quite repeatable across the three interpretations, but that polygons are less consistent. Different options for constructing and maintaining temporal histories are discussed. Fuzzy space-time composites are one option, while the use of transition polygons in pairwise overlay supports an iterative or dynamic update of forest maps, where only the current state of the forest is accessed at any one time. The fuzzy space-time composite of the three forest maps may also be used as a fairly complete base map portrait of the forest for incremental updates in the future.

1. INTRODUCTION

The problem of managing spatial data over time has become increasingly important over the past several years. Many application domains which have embraced GIS technology have found their capacity of adaption to handling temporal data to be extremely limited. Existing GIS must represent temporal sequences of spatial data by resorting to procedures which involve heavy data redundancy or which are extremely slow. These difficulties are further exacerbated by the presence of spatial uncertainty in most maps. When maps of different epochs are overlaid, severe polygon fragmentation usually results. New developments are beginning to emerge which do not require as much redundancy and which may be faster for certain kinds of querying (e.g. Gold, 1992). However, the problems of fragmentation in the presence of spatial uncertainty must also be dealt with if such solutions are to be useful for managing real world data.

Polygons in many domains are determined via interpretation of aerial photos. This is true, for example, in forestry, soil science, and geomorphology, agriculture, geology, urban planning, hydrology, hydrography and many other disciplines. Sometimes, polygons are acquired via interpretation or classification of satellite imagery, but the latter are usually validated with maps generated from photointerpretation. While sharp boundaries such as buildings, parking lots, roads, etc. may be determined reliably from photointerpretation, most boundaries of interest derived from this data source involve either intrinsic uncertainty or are heavily generalised during the extraction process. Hence, for example, coastline maps will invariably be produced with some generalisation, while forest and soils boundaries are examples where boundary placement is of high intrinsic uncertainty.

Interpretation of aerial photographs in these application domains is further characterised by the widespread use of a "minimal mapping unit". This refers to the necessity to group small regions of sometimes distinctly different classes within larger regions because they are considered to be "too small" to map directly (or the interpreter has been instructed to consider them too small). When combined with regions containing cover type mixtures which are difficult to discriminate one from another, such as in forestry, the use of a minimal mapping unit tends to increase the spatial (and categorical) uncertainty in the interpreted map.

Only relatively recently have attempts been made to compare maps electronically which were produced at different epochs or by different methods (e.g. Chrisman 1989). Maps produced via photointerpretation compare poorly. Estimates of map accuracy from photointerpretation (carried out in comparison to ground surveys) indicate error can be as high as forty to fifty percent (Biging *et al.* 1992). New maps are produced without any attempt to align the new interpretation with previous interpretations of the same territory. The result is that polygons cannot be compared directly from one date to the next, because they represent different aggregate groups of smaller regions. Even when change has been slight, the differences between the maps suggest change has been high. This presents great difficulties for understanding the evolution of regions over time.

In this paper, new methods for characterising both boundary error and for ensuring the correspondence of spatial entities (polygons) from one map date to another, over a sequence of map dates, are presented. Previous work by the author and colleagues (Aubert *et al.* 1994; Edwards 1994) on characterising boundary error is summarised. This is an important step, for both the boundary uncertainty and the heterogeneity inside polygons contribute to polygon incoherence, and the one can easily be confused with the other. We have found it easier to develop methods for handling boundary uncertainty first (Aubert *et al.* 1994; Edwards 1994), and have dealt with heterogeneity and the related issue of polygon aggregation as they relate to temporal cohesion at a second level (this paper). Issues such as the fragmentation of the spatial database are also discussed in this context.

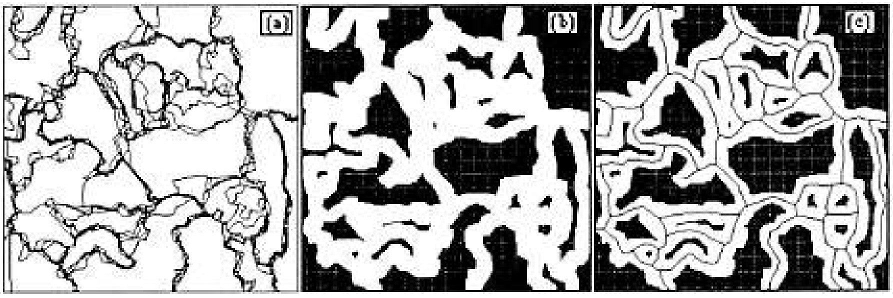


Figure 1. (a) Four interpretations of a single image; (b) A four pixel corridor for the set of four interpretations; (c) A sketch illustrating the skeletons of the corridors.

2. BOUNDARY UNCERTAINTY

Boundary uncertainty has been addressed in several parallel research projects, each of which shares the same core method but addresses a different application. The central work has been developed around several artificially textured images which have been interpreted independently by a large number of interpreters (up to thirteen in total). These multiple interpretations of the same scene have been analysed under a raster implementation in order to extract measures of the uncertainty width (or fuzzy width) of each boundary (Aubert *et al.* 1994). This uncertainty width is simply the distribution of several interpreted boundaries around an estimate of the mean interpreted boundary, and hence may be viewed as a quantification of interpretation uncertainty. The method used in this project consists of overlaying several interpretations of the same scene, calculating a set of corridors in order to group the interpretated boundaries, removal of outliers, determining the skeleton of each corridor after outlier removal has been completed and determining the width of each corridor (correcting for the corridor width introduced earlier on) as a measure of boundary width (see Figures 1 and 2). Hence the method developed leads to a unique width for each pair of adjacent polygons and their common boundary, for a given set of multiple interpretations.

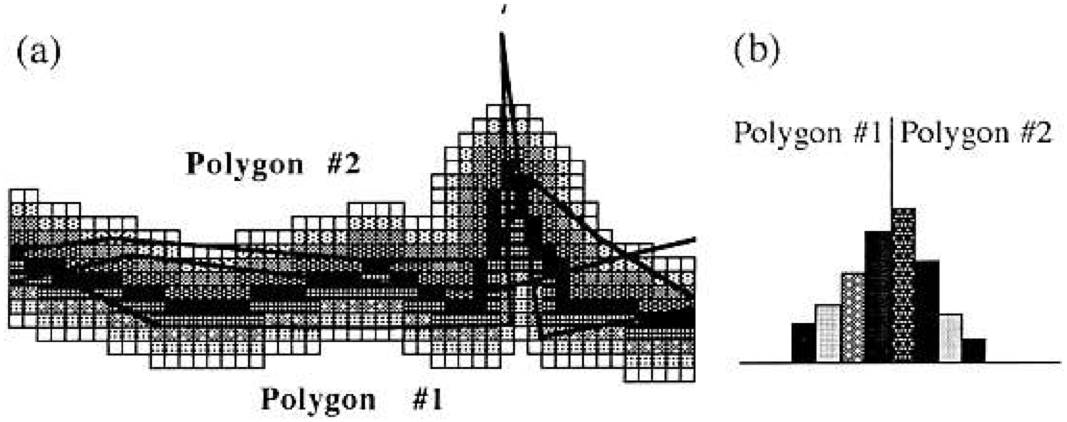
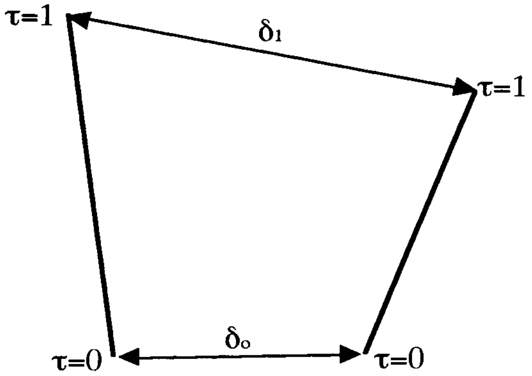


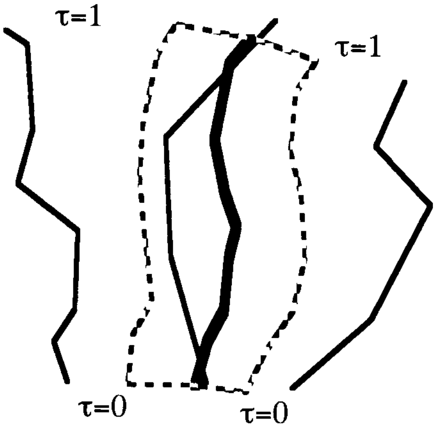
Figure 2: The raster method used to extract a boundary profile from the set of multiple interpretations and the known mean. (a) Corridors are genenerated around the mean boundary and the sets of multiple interpretations are then overlain on the corridors. Whenever a interpretation crosses a pixel belonging to a given corridor differential, the count for this distance from the mean is accumulated. The resulting histogramme is shown in (b).

A vector implementation of this method has been developed which favours the use of a statistical framework based on clustering theory to determine when two boundaries are similar enough to be treated as the same or not (Edwards 1994). The method is based on a parametric characterisation of boundary segments and a corresponding curve-to-curve distance measure (see Figure 3). Briefly, the curve-to-curve distances are used as a measure of curve similarity to group the most similar curves from different interpretations. For each cluster formed in this way, a mean boundary and a one-sigma "box" are determined (Figure 4). When the mean boundary segments formed in this way are reconnected into a boundary network, they produce a kind of "fuzzy overlay" of boundaries, where boundaries which were "nearby" have been grouped together and only boundaries which were "isolated" have maintained a separate identify. This fuzzy overlay is different from methods designed to remove sliver polygons, in two key ways — (i) each boundary in the overlay is associated with a fuzzy width, and (ii) none of the original boundaries have been "moved". The fuzzy overlay is a new map, generated from the old ones without arbitrarily moving any of the line features of the latter. This method has been applied to both map overlay and map update, to show that the result of the "overlay" removes fairly naturally all sliver polygons as well as providing a measure of boundary uncertainty.



$$D = \text{minimum}\left\{\left[\int_{\tau=0}^{\tau=1}\delta(\tau)\,d\tau\right],\left[\int_{\tau=0}^{\tau=1}\delta(1-\tau)\,d\tau\right]\right\}$$

Figure 3: The determination of the curve-to-curve distance function. The two curves are parameterized between 0 and 1, and the Euclidean distance (or any other distance function) as a function of the parameter τ is integrated along the length of the curves. The distance function is the minimum of the two possible distances, depending on the orientation of the two curves.



$$D_+ = \sum_{j=1}^{N_{segs}}\left[\int_{\tau=\tau_{0,j}}^{\tau=\tau_{1,j}}\delta(\tau)\,d\tau\right] \quad \text{where } \tau_{0,j+1}=\tau_{1,j}$$

Figure 4: This illustration shows the curve-to-curve distance generalised to polylines. The heavy line near the centre represents the mean curve, and the dotted line a representation of a one-sigma box. The formula shows the distance function for one orientation, where N_{segs} represents the number of line segments on the mean polyline, and is less than or equal to the sum of line segments of each member line less one.

The vector method is presently rather too labour intensive for practical use. The difficulty resides in the initial segmentation of the interpretation boundary networks into matched pairs of polylines. This phase of the work is presently handled largely via human intervention, although a toolkit of functions has been developed to facilitate this work. A more automated technique for carrying out the initial boundary segmentation appears to be feasable, but more work is required to achieve this. The raster method is therefore, for the moment, the more promising approach. It is hampered, however, by a number of inconveniences. It requires a corridor width to be specified, and the results are sensitive to the choice of this corridor width. Secondly, considerable effort is required to develop the code to carry out the corridor analysis, outlier removal, the skeletonisation and the binning to obtain boundary profiles. Until now, this has been carried out manually within existing image analysis software (ERDAS), but this is not appropriate for practical use. Finally, a lot more evaluation of the effects of different choices on the raster analysis needs to be carried out before these techniques are used operationally. Also, the two methods (raster and vector) are based on different concepts and produce different results. These differences have not yet been quantified, but their existence suggests that more study is needed to understand them. For this paper, the vector method was used, because, for the moment, it is the most fully implemented.

Finally, a third project has been devoted to an effort to quantify the relationship between boundary uncertainty as determined from the multiple interpretations and the properties of the textured patterns on either side of each boundary (Edwards and Lowell 1994). Boundary uncertainty in this work may be understood at representing the repeatability of different interpreters finding the same boundary at the same location. During the course of this work, a fairly simple theory was proposed for the relationship between image properties and the interpreted boundary uncertainty (Figure 5). Specifically, the width of the uncertain boundary region was shown to be a function of three factors: (i) pattern discrimination (i.e. the patterns on either side of the perceived boundary) (ii) the scale of the variability of the patterns and (iii) a variety of context effects.

The relationship between pattern discrimination and boundary uncertainty is fairly clear. If the patterns on either side of the perceived boundary are too similar, boundary uncertainty will be high and the width of the transition zone will be found to be large.

The scale of the pattern variability or heterogeneity has a somewhat more subtle effect (Figure 6). An interpreter attempting to locate the border between two patterns with different scales of heterogeneity will find that the location of the interpreted boundary will be affected by one or both of the characteristic scales of the patterns on either side of the boundary. If one of the texture patterns is more uniform in appearance than the other, then the more variable pattern's scale will dominate the uncertainty in the boundary placement.

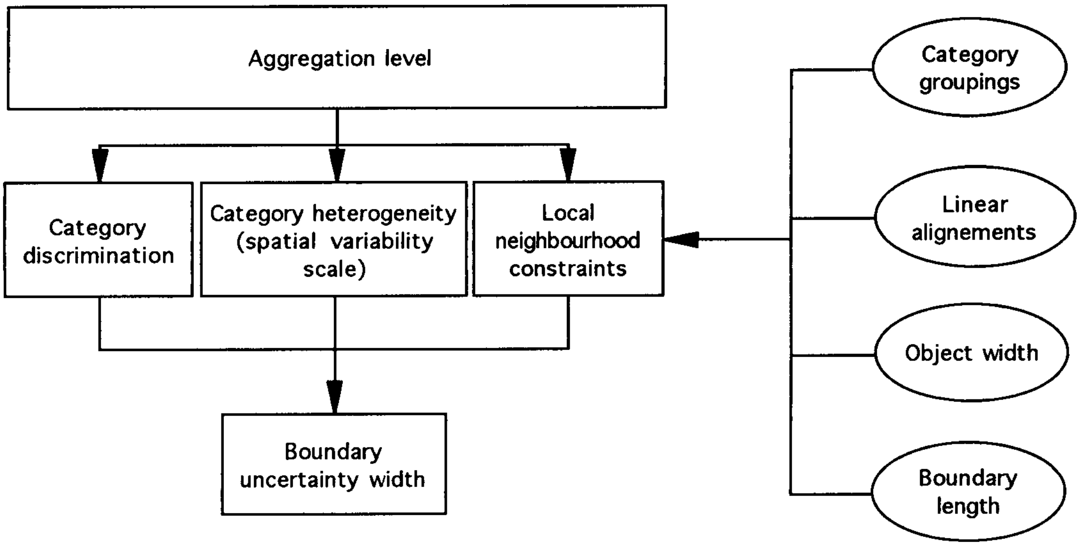


Figure 5: A schematic view of the functional relationships between different variables related to boundary uncertainty.

Finally, context effects may be of several types. The length of common boundary between two patterned regions affects the ability of the interpreter to see the distinction. The width of the polygons orthogonal to the boundary also affect the perception of the boundary. Other shape effects also appear to be present. In particular, any boundaries which form a linear alignement may be located with lower uncertainty than more curvilinear features. The presence of similar patterns in the neighbourhood may also affect discrimination capability. The theory was found to be successful for three different artificially textured images. It is also similar in structure to current theories of human perceptual processing.

As shown schematically in Figure 5, the level of aggregation which has been decided beforehand will affect all three sources of boundary uncertainty. Hence if several kinds of similar patterns are grouped into one category, heterogeneity will be higher and the scale of heterogeneity may also be increased. Furthermore, context aspects such a shape will also be affected by grouping polygons. On the other hand, discrimination between the grouped category and others may be improved, hence decreasing boundary uncertainty.

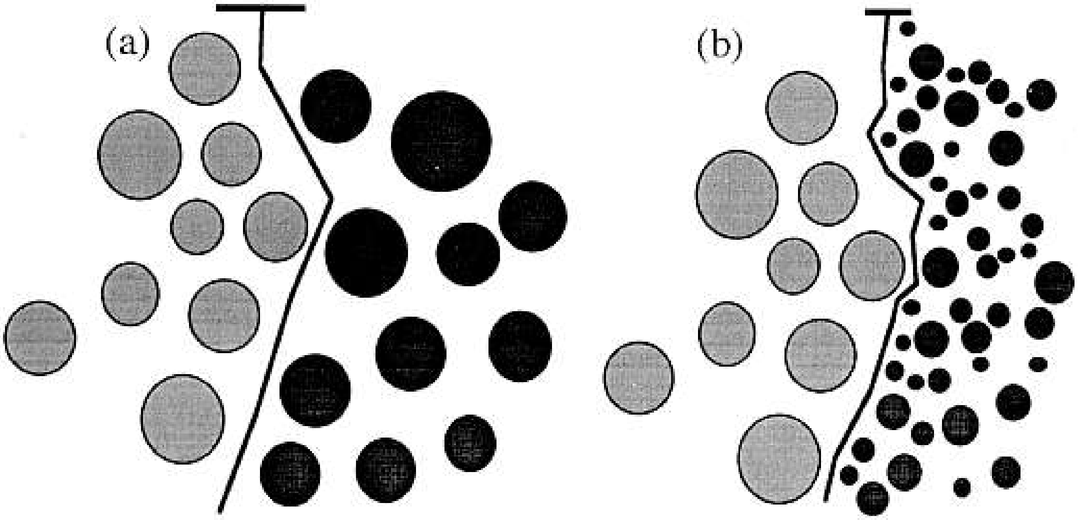


Figure 6: The scale of pattern variability affects boundary uncertainty in different ways. Shown in (a) is the effect of variability scale on boundary uncertainty for two patterns with similar scales, while in (b) is seen the case of two patterns with different variability scales.

3) ESTABLISHING POLYGON COHERENCY OVER TIME

The developments described in the previous section are based on the analysis of more than one photointerpretation for each date for each region. However, it is possible, with some modification of the techniques, to use them for following a time sequence of maps. Such a sequence of three maps (1970, 1980 and 1990) for a small part of an experimental forest belonging to Laval University, the Montmorency Forest has been used for this study. The maps used were the standard forest inventory maps produced by the Québec province's Ministry of Natural Ressources, at a scale of 1:20000. In Figure 7a, these three maps are directly superimposed using standard map overlay procedures. The results, as should be clear, are not very useful for determining which polygons persist over time.

Fuzzy overlays were computed using the vector techniques outlined in (Edwards 1994) and described in the previous section. Fuzzy overlays were determined, first for the map pair consisting of the 1970 and the 1980 maps, and then for the map pair consisting of the 1980 and 1990 maps. These two fuzzy overlays were then combined into a single fuzzy overlay (Figure 7b).

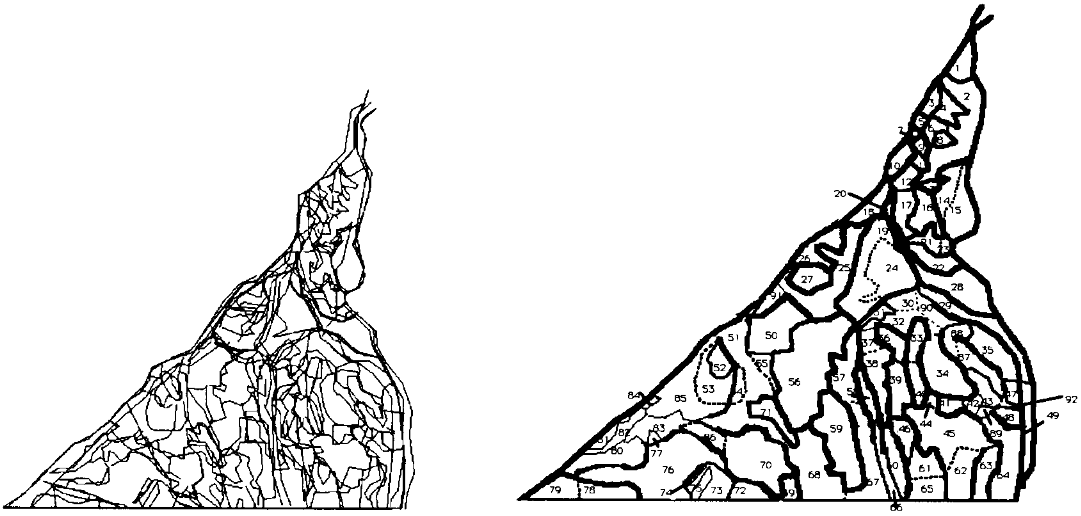


Figure 7. (a) The standard overlay of all three maps; (b) The fuzzy overlay of all three maps. ▬ represent boundaries common to all three epochs. ▬ represent boundaries common to any two epochs. ▪▪▪▪▪ represent boundaries only present in 1970. — represent boundaries only present in 1980. --- represent boundaries only present in 1990.

This overlay contains the full set of polygon primitives which must be defined over the sequence of three dates in order to ensure coherency in the polygons. The overlay was produced from the first two overlays by (i) first identifying boundary segments common to all three dates; (ii) identifying boundary segments common to temporally adjacent dates; (iii) identifying boundary segments common to the first and last date; (iv) identifying boundary segments unique to each date; and (v) for unique boundaries, classifying them into boundaries which represent a strong attribute contrast and boundaries which represent a weak attribute contrast. Also roads, streams and rivers were suppressed in the production of the final overlay.

There were 28 polygons present in 1970, 38 polygons in 1980 and 39 polygons in 1990. The combined set of polygons for all three dates contains 92 polygons. This should be compared to more than 1000 polygons if the overlay is not carried out using the fuzzy boundary procedure (e.g. Figure 7a). However, of the full set of 92 polygons, only 68 polygons were needed to make the transition from 1970 to 1980, while 73 polygons were needed for the transition 1980 to 1990. The interpretation norms used to define the 1970 map were coarser than those used for either the 1980 or 1990 map, hence the lower number of polygons.

Boundaries in Figure 7b due to a single map are, in general, the result of a clearcut, of other real changes in the forest database, or of the presence of subtle contrast effects not seen by other interpreters. Boundaries common to two maps correspond to smaller units which emerge when analysing multiple interpretations. Boundaries present on all three maps form larger aggregate units which are easier for photointerpreters to find in the images. Hence, for example, the attribute contrast between polygon #66 and polygon #60 is low and these two regions could easily be grouped together by a different interpreter. The same is true for polgyons #50 and #56. On the other hand, the apparent growth of polygon #52 to embrace polygons #53 and #54 in the 1990 map may represent real change.

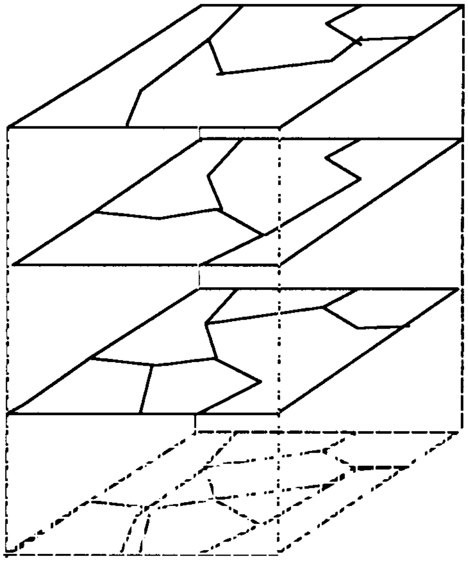


Figure 8: The emergence of smaller polygons from a sequence of interpretations based on larger polygons which are aggregates.

The preceding analysis shows that, somewhat contrary to expections, the majority of boundaries of forest stands were found to be present on at least two of the three maps acquired at ten year intervals. This result was surprising for several reasons. Experienced foresters indicate that one cannot expect forest maps generated at ten year intervals, using different inventory standards, to be comparable. Yet in this sample (and admittedly small) region, the stand boundaries are largely coherent. Secondly, previous unpublished work on the same data set suggested that at least a third of polygons should be incoherent from one date to the next. In actual fact, the results described here are probably consistent with this observation. Although most boundaries are found to be coherent, there are enough boundary segments which are not that they affect about a quarter or more of the polygons. Hence, if our criteria for coherency consists of polygons, then there is little one-to-one coherency from one date to the next. If, on the other hand, we focus our attention on boundary coherency, we find that over 80% of the boundaries are coherent between at least two of the three maps.

The boundary segments present at only one date represent either real change or border regions whose categories are easily confused. Indeed, the same observation holds true for boundaries present on two dates but not on all three. The presence of heterogeneity in the textures on which the categories are based, and the role of the minimum mapping unit as discussed in the first section, mean that the larger polygons which are interpreted as stands are made up of a number of smaller polygons, sometimes with quite distinct categories, which are grouped together by different interpreters in different ways at different times. When comparing any two interpretations, one is struck by the large numbers of boundaries on one map which are not present on the second. However, when three or more interpretations are combined, more consistency emerges in the boundaries. Indeed, these smaller polygons emerge as coherent units with boundaries which, in general, belong to at least two maps, although different sections of the boundary of such a smaller polygon may not necessarily belong to the same two maps (see Figure 8). The analysis above shows that, over three forest maps, isolated boundary segments are rare. It was expected that such isolated boundary segments would be more common in real forest maps. Earlier work (Aubert *et al.* 1994) showed that, for the artificial images which were intepreted in the first stage of the work, three interpretations contained over 80% of the boundaries found in nine interpretations, suggesting that three interpretations contains almost all the information. The new result is consistent with this earlier result.

4. IMPLICATIONS FOR SPATIAL-TEMPORAL DATABASES IN THE PRESENCE OF SPATIAL UNCERTAINTY

The presence of smaller, but coherent units in the combination of several maps means that polygon coherency can be determined over time using these smaller regions. There are two ways to do this. The first is to treat the smaller polygons as real and then to construct temporal histories for each one. The approach is similar to the space-time composite described by Langran (1992), except that the map overlay carried out here is a fuzzy overlay, not a standard overlay. The difficulty with this technique is that the accumulation and integration of maps at more dates may result in an increasing fragmentation of the database. A second difficulty is that, although polygon histories in the space-time composite will be different, two adjacent polygons may have very similar and sometimes identical categories at any given time. This contradicts standard practice, which assigns a unique identifier only to polygons representing different categories, and hence would require redefining how polygons are assigned identity.

The "fuzzy space-time composite" which results from this fuzzy overlay procedure may be interpreted differently, however. It is more complete than any of subset of its constituent maps and hence represents a fairly good portrait of the forest. Some temporal change effects are incorporated into its structure, but most polygon distinctions represent category differences. The composite also contains a great deal of information on boundary uncertainty. By combining the maps from the earlier three dates, therefore, it is possible to build a single portrait of the forest which could be used as a "base map" description of the forest. Using such a map, it should no longer be necessary to acquire independent forest maps every ten years. Rather, the next forest inventory should be carried out by interpreters using the existing fuzzy map and looking only for change.

An alternative approach is to treat the smaller polygons as "transition" polygons, that is, polygons which are used only to make the transition from one date to the next, and which are then discarded. This approach is valid when the transition polygons really are transitory — that is, they do not reappear again during later transitions. If the latter is true, then their inclusion directly in the database as distinct entities makes more sense. It is also valid only if there is a mechanism for iteratively updating the database, destroying earlier polygons and making new ones. If all polygons are incorporated directly into the database, then there is no justification for using transition polygons. However, given these two conditions, the use of such transition polygons ensures that the database does not fragment into smaller and smaller pieces.

If we are to build such a spatial-temporal database, the analysis of the preceding section indicates that we would need three map databases, each containing roughly 40 polygons, and two transition databases containing transition rules for around 70 polygons. The latter are subdivisions of the single date polygons required to carry out the comparison from one date to the next. These transition polygons could be computed using fuzzy overlay procedures, but could not be generated using standard overlay. Presently the fuzzy overlay procedure must be carried out interactively and hence there is too much overhead in time and effort associated with the procedure for it to be carried out on the fly. It is therefore better to determine the fuzzy overlays ahead of time and built them directly into the database.

5. DISCUSSION AND CONCLUSIONS

In this paper, a procedure is outlined using newly developed fuzzy overlay techniques to compare three forest maps based on aerial photos acquired at different dates over ten year intervals. It was found that most boundaries could be identified on two or more of the maps. Exceptions include boundaries which represent real change and highly confused category pairs, but these latter were rare. Hence boundary coherency across three interpretations is high, although polygon coherency was found to be lower. Two methods of using the fuzzy overlay techniques to build a spatial-temporal database were discussed. One consists of producing a fuzzy space-time composition, the other of using transition polygons in a dynamic update procedure. The choice of approach depends on the available technology as well as the nature of the entities being tracked over time. It was also pointed out that the composite fuzzy map produced by combining the three independent forest maps contains more information than any subset of its component maps, and that the composite map may actually be used as a more complete map of the existing state of the forest. The use of such a composite map might relieve the need for redoing the entire forest map the next time around (in five to six years).

6. ACKNOWLEDGEMENTS

The work outlined in this paper was financed by the Association des industries forestières du Québec and the Canadian NSERC, with contributions from the Québec Ministry of Natural Ressources and Rexfor.

7. REFERENCES

Aubert, E., G. Edwards and K.E. Lowell, 1994, "Quantification des erreurs de frontières en photo-interprétation forestière pour le suivie spatio-temporel des peuplements", *Canadian Conference on GIS*, Ottawa, Ontario, 195-205.

Biging, G.S., R.G. Congalton and E.C. Murphy, 1992, "A Comparison of Photointerpretation and Ground Measurements of Forest Structure", *Proceedings of the ASPRS Annual Meeting*, Baltimore, Maryland, Volume 3, 6-15.

Chrisman, N.R., 1989, "Modeling error in overlaid categorical maps", *Accuracy of Spatial Databases*, Michael Goodchild and Sucharita Gopal (eds), 21-34.

Edwards, G., 1994, "Characterising and Maintaining Polygons with Fuzzy Boundaries in Geographic Information Systems", *Proceedings of the 6th Symposium on Spatial Data Handling* , Edinburgh, Scotland, 223-239.

Edwards, G., and K.E. Lowell, 1995, "Modeling Uncertainty in Photointerpreted Boundaries", to appear in *Photogrammetric Engineering and Remote Sensing.*

Gold, C.M., 1992, "Dynamic Spatial Data Structures — the Voronoi Approach", *Proceedings of the Canadian Conference on GIS,* Ottawa, 245-255.

Langran, G., 1992, *Time in Geographic Information Systems*, Taylor and Francis, New York, 189 pp.

A SYNTAX FOR THE REPRESENTATION OF FUZZY SPATIAL OBJECTS

MARTIEN MOLENAAR
Centre for Geo-Information Processing (CGI)
Wageningen Agricultural University
The Netherlands
E-mail: MARTIEN.MOLENAAR@WETENSCH.LMK.WAU.NL
Tel: +31(0)8370-82910 Fax: +31(0)8370-84643

ABSTRACT.

Terrain objects will always be defined within some users context and these definitions will often have a certain level of vagueness or fuzziness. Data models for the representation of such objects in information systems should be able to deal with the uncertainty of their thematic and spatial description. The definition of a syntax for these data models will help to understand:

- how thematic and geometric data can be related to the represented objects,
- how the spatial aspects of the objects can be represented in different geometric models and what the similarity of these models is,
- how uncertainty can be taken care of in the object descriptions

The uncertainty of the object descriptions will propagate to the spatial relationships between the objects, this will demonstrated for some examples.

1. INTRODUCTION

The objects represented in a geo-information system will always be conceptual entities that play a role in some terrain description. This role is made up by the different relationships they have with other objects and their behaviour in time under external influences or due to intrinsic factors. Consequently one should understand the object definitions in the context of such a role pattern and behaviour pattern, this context will have several aspects (Molenaar, 1993). The first aspect is the discipline or disciplines of the users, i.e. are they working in a cadastral environment, or soil mapping, or demography etc. Each discipline will have its own definition of terrain objects, classes and attributes. These definitions also depend on the scale level or the aggregation level of the mapping, i.e. will the mapping be made at a local level, a regional level, a national level or even a continental level; e.g. at a municipality level a GIS may contain houses, streets and parks, while at a national level a GIS may contain towns and urban areas. Another aspect of the users context is the aim of the mapping, i.e. will the data be used for monitoring the terrain situation, or for the identification and analysis of processes, or will the data be used for planning purposes etc. A fourth important aspect of the users context is the time of the terrain description. In many disciplines and users environments the relevance of data changes with time. In agriculture for example, the requirements for soil information have changed during the last decades. Originally the main interest was to analyze the suitability of soils for different crops. At present the interest changes to e.g. the capacity of the soil to bind chemical elements which could harm the environment. In the cadastral world we see that, whereas the original tasks were to protect land titles and to raise taxes, they now often play an increasing role in the analysis of economic processes, such as the dynamics of prices of real estates and of the number of sales and mortgages.

These considerations imply that the objects represented in a geo-information system are not simply entities that are "out there", but they are rather representations of conceptual entities that play a role in some descriptive model of the earths surface. Such a model defines the semantics of the objects and their role patterns. Each users context will have its own semantics. Many applications have some kind of indeterminacy in their descriptive models in the sense that it is not always clear how the conceptual entities in the model could be linked unambiguously to real world phenomena. This might be due to the fact that the measuring procedures for data acquisition have stochastic components, but it might also be that the data categories of the model have fuzzy definitions, or that inferences are based on insufficient data. This means that uncertainty will always be an intrinsic aspect of information so that if data models are developed for geo-information then they will not be complete if they are not able to deal with uncertainty.

A terrain description contained in a GIS will be called a map in this paper, thus "map" refers to the data contained in a geo-database rather then to a graphical display of such data. If the geometry of a map M has vector structure then M will be called a *vector map* and similarly if the geometry has a raster structure then M will be called a *raster map*. There are many other types of tessellations for a two dimensional space but we will restrict our discussion to these two. Many of the statements made in later sections about the comparison of the raster and vector geometry can be extended to those other tessellations too.

For our further developments we will assume that representations of terrain objects in a geo-information system have the basic structure of figure 1. This representation consists of an object identifier to which the thematic data and the geometric data are linked. With the notation developed in section 2.1. it will be possible to explain that vector maps and raster maps do have basically the same syntax. This notation will also help us to understand how uncertainty can expressed when semantics are introduced into the model.

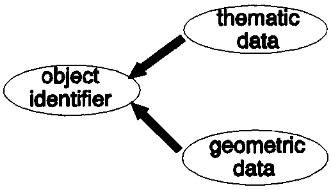


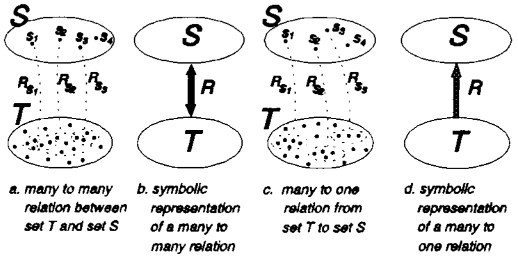
fig. 1: The basic structure for representing terrain objects in GIS.

A two dimensional terrain description could contain point-, line- and area objects. The following elaborations will be restricted to the representation of line and area objects. A further restriction with respect to the concept of objects as found in modern computer science is that object dynamics or object behaviour will not be discussed here and no attention will be paid to composite or aggregated spatial objects. The main aim of this paper is to formulate a syntax that can handle the semantics of state descriptions of fuzzy spatial objects. No explicit attention will be paid to the time or scale dependency of the object states, we rather take it that uncertainty due to these aspects can somehow be expressed as uncertainty of the object state as such. The basic concepts of the syntactic approach can be sufficiently explained under these restrictions.

2. A SYNTAX FOR OBJECT REPRESENTATIONS

2.1. *Many to One Relations Between Sets*

A conceptual data model identifies a number of data types and relations between these data types. A syntax for such a model will be formulated in a mathematical sense, the data types will be interpreted as discrete sets between which relations will be defined (Gersting,1993). This model is graphically represented by figure 2. Each ellipse represents a set identified by the label; an ellipse labelled by S represents a set with elements s_i, i.e. $S = \{...., s_i, ...\}$ and an ellipse labelled with T represents a set $T = \{..., t_j, ...\}$. In figure 2a the elements of subsets of T are related to elements of S through some binary relation $\Re \subset T \otimes S$. Each subset will be labelled with a symbol R identifying the type of relation that generates the subset and an index s_i identifying the generating element or owner in set S. For some subsets R_{Sk} and R_{Sl} we might find that $R_{Sk} \cap R_{Sl} \neq 0$, in that case $\Re$ is a *many-to-many relation* because the subsets of T are overlapping, see figure 2a. This situation has been represented symbolically in figure 2b where the two headed arrow means that the relation is many-to-many.



a. many to many relation between set T and set S
b. symbolic representation of a many to many relation
c. many to one relation from set T to set S
d. symbolic representation of a many to one relation

fig. 2: Many-to-many and many-to-one relations between two sets.

Let $R = \{..., R_{Si}, ...\}$ be the collection of all subsets of T generated by $\Re$, if all these subsets are disjoint, i.e. if for all combinations of subsets $R_{Sk} \cap R_{Sl} = 0$ then the relation $\Re$ is *many-to-one* from set T to set S. This situation has been illustrated in figure 2c. and has been represented symbolically in figure 2c; the arrow represents the fact that the relation is many-to-one from set T to S, this means that in fig. 3d $\Re$ relates 0 or more elements of set T to each element of S (see (Ullman,1982), (Molenaar,1991a and b)). If an element $s_i \in S$ generates a subset $R_{si} = \{t_p, ..., t_q\} \subset T$ then s_i is the owner of R_{si}, therefore the set S is called the owner set of R. The elements of T are members of the subsets generated by R therefore T will be called the member set of R. For the subset R_{si} a membership function will be defined so that:

$$R[t_k, s_i] = 1 \leftrightarrow t_k \in R_{si}$$
$$R[t_k, s_i] = 0 \leftrightarrow t_k \notin R_{si}$$

The fact that for $s_i \neq s_j \rightarrow R_{si} \cap R_{sj} = 0$ implies that

$$R[t_k, s_i] = 1 \rightarrow R[t_k, s_j] = 0$$

The choice of this notation for membership functions was made for two reasons: they can be used for the computation of cardinality numbers of sets, they can easily be adapted for the situation where we have to deal with fuzzy sets (Klir and Folger 1988) and they can be used to compute membership functions for the composition of relations. These relationships could alternatively expressed as predicates, or as (labelled) binary relationships. In the latter form they can be compared to the regular entity relations of (Chen,1976).

2.2. *Object Classes*

Let U_M be the set of all objects occurring in a map M, then U_M is called the *universe of discourse* of M, or shortly *the universe* of M. Objects in this universe can be distinguished because they have different characteristics, for most GIS applications these differences will be primarily thematic. A thematic description of the objects is then required to express the differences, this can be done by means of attributes that take values per object. The complete description of an object in a GIS consists then of the thematic component expressed by the attribute values and a geometric component. Two objects can only be distinguished if their descriptions are not equal.

Class intension and extension

Carnap explains how classes are defined by the properties that are characteristic for the members, i.e. the objects of a class have properties that distinguish them from objects that do not belong to the class (Carnap,1956). This implies that for each object of a universe U_M a decision must be made to assign it to the class or not. Criteria should therefor be formulated for each class to specify which objects should be considered as class members. These criteria express the *intention* of the class and they can be operationalised by a decision function that can be applied to the objects to test whether they fulfil the criteria or not. If an object O_i passes the test for a class C_j then it will be a member of that class and that will be expressed by the membership function:

$$M[O_i, C_j] \quad = 1 \text{ if } O_i \text{ is a member of } C_j$$
$$= 0 \text{ otherwise.}$$

For each object O_i we define the function specifying the classes to which the object belongs:

$$CLASS(O_i) = \{C_j \mid M[O_i, C_j] = 1\}$$

If the classes are defined so that they are mutually exclusive then for each object the extension of this function contains only one class.

The *extension* of a class is the set of all the objects that belong to it, hence:

$$EXT(C_j) = \{O_i \mid M[O_i, C_j] = 1\}$$

It is possible that a class C has been defined to which no objects belong, then the extension is empty, i.e. $EXT(C) = \varnothing$

Objects, Classes and Attributes

The general situation in a terrain description is that there are sets of objects that are so much different that this can not any more be expressed by the attribute values only. Each set should then have its own description structure, i.e. each set should have its own list of attributes; these sets will be called *object classes* or short *classes*. The classes are typified by the fact that the objects that belong to the same class share the same descriptive structure. A class C_j has then a list of attributes:

$LIST(C_j) = \{A_1,..,A_r,.., A_n\}$

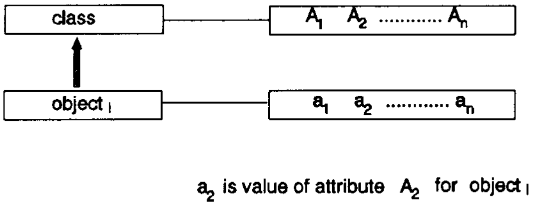


fig. 3: Diagram representing the relation between objects, classes and attributes

Each attribute will have a name and for each attribute a domain will be specified defining the complete set of attribute values. The attribute structure of an object is determined by the class to which it belongs, so that each object has a list containing of one value for every attribute of its class. An object inherits the attribute structure of its class, i.e.
$M[O_i, C_j] = 1$ implies that :

$LIST(O_i)=\{a_1,..,a_r,..,a_n\}$
with $a_r = A_r[\,O_i\,]$ *is value of* A_r *for object* O_i
$A_r \in LIST(C_j)$.

The diagram of figure 3 illustrates the relationship between objects and classes and the attribute structures. The thematic description of an object can now be specified by the class of the object and the list of attribute values, the class specifies the names of the attributes. Classes are semantically distinct if their attribute lists are different, i.e.

$i \neq j$ *implies that* $LIST(C_i) \neq LIST(C_j)$

We will assume that this is the case, this means that objects belonging to different classes have different descriptions. In the relational database model this would mean that a table can be defined for each object class.

2.3. The Geometry of Line and Area Objects

The Representation of Line and Area Objects in a Vector Map

The geometric structure of a vector map can be described by means of cell complexes. For a two dimensional map these consist of 0-cells, 1-cells and 2-cells (Frank and Kuhn,1986). The 0-cells and 1-cells play similar role as respectively the nodes and edges when the geometry of the map is interpreted as a planar graph, the 2-cells can then be compared to the faces related to the planar graph through Eulers formula (Gersting,1993). The terminology of the planar graph interpretation will be used here.

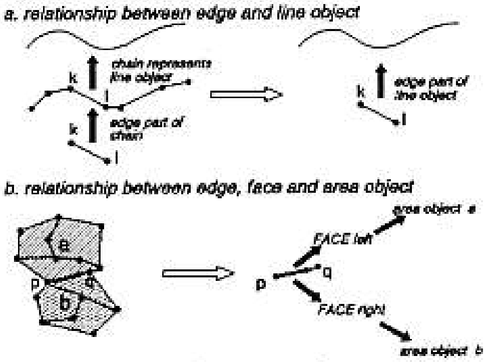
The geometry of a simple line object is represented by an chain of edges as in figure 4a. If an edge e_i is part of such a chain representing a line object O_l then the relation between edge and object will be expressed by the function

$PartLO[e_i, O_l] = 1$

This function will have the value = 0 if the edge is not part of the chain.

The geometry of an simple area object is represented by one or more adjacent faces as in figure 4b. If a face F_f is part of an area object O_a this will be represented by:

$PartAO[F_f, O_a] = 1$


a. relationship between edge and line object

b. relationship between edge, face and area object

fig. 4: Relationships between edges and objects.

If there are overlapping area objects then each face might be part of several objects, but also each object will consist of one or more faces. Therefore this is a many-to-many relationship. Because all area objects are related to faces this is a mapping from the set of area objects to the set of faces. Overlapping objects can be found through the common faces.

The relationships between edges and area objects can be found through their relationships with the faces in combination with the relationship between the faces and the area objects. Each edge will always have one face at its left hand side and on at its right hand side. These relationships will be expressed by the following functions:

- Edge e_i has face F_f at its left-hand side → $Le[e_i, F_f] = 1$
  For any $F_h \neq F_f$ we get then $Le[e_i, F_h] = 0$
- Edge e_j has face F_f at its right-hand side → $Ri[e_j, F_f] = 1$
  and again for $F_h \neq F_f$ we get then $Ri[e_j, F_h] = 0$

If an edge e_i is part of the border of F_f then only one of the functions Ri and Le is equal to 1, but not both. So if we define the function:

$B[e_i, F_f] = Le[e_i, F_f] + Ri[e_i, F_f]$

then when e_i is part of the boundary of F_f we find $B[e_i, F_f] = 1$.

Now it is possible to check whether edge e_i is related through face F_f to an area object O_a Therefor the following function should be evaluated:

$Le[e_i, O_a \mid F_f] = Le[e_i, F_f] \times PartAO[F_f, O_a]$

This function takes the value 1 if both functions at the right hand side are = 1, i.e. the edge should have the face at its left hand side and the face should be part of the area object. In all other cases the function at the right hand side of the equation takes the value = 0. The fact whether the edge is related to the object can then be found through:

$Le[e_i, O_a] = MAX_{Ff}[\, Le[e_i, O_a \mid F_f] \,]$

There is at most one face for which both $Le[e_i, F_f] = 1$ and $PartAO[F_f, O_a] = 1$, if such a face exists then the function relating the edge to the object will get the value = 1, in all other cases the product will be = 0. Hence if edge e_i has area object O_a at its left-hand side $\rightarrow Le[e_i, O_a] = 1$. Similarly we can write:

$$Ri[e_i, O_a \mid F_f] = Ri[e_i, F_f] \times PartAO[F_f, O_a] \text{ and}$$
$$Ri[e_i, O_a] = MAX_{Ff}[\, Ri[e_i, O_a \mid F_f]\,]$$

if edge e_i has area object O_a at its right-hand side $\rightarrow Ri[e_i, O_a] = 1$, otherwise it is = 0.

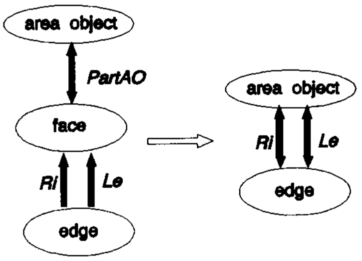


fig. 5: Transition from edge-face-object relationships to edge-object relationships

The transitions from the indirect relationships between edges and area objects to the direct relationships have been represented the diagram of figure 5, where the many-to-many relations are represented by double headed arrows. The relations between faces and area objects are many-to-many, so that will also be true for the derived relations between edges and faces. The combination of these two functions gives for edge e_i :

$$B[e_i, O_a] = Le[e_i, O_a] + Ri[e_i, O_a]$$

If an edge e_i is part of the border of O_a then only one of the functions Ri and Le is equal to 1, but not both. So for such an edge we find $B[e_i, O_a] = 1$, the boundary of O_a consists of the set of edges with

$$\partial_C O_a = \{ e_i \mid B[e_i, O_a] = 1 \}$$

If e_i has O_a both at its left-hand side and at its right-hand side then $B[e_i, O_a] = 2$, in that case it is running through O_a. If $B[e_i, O_a] = 0$ there is no direct relationship between e_i and O_a (Molenaar,1991a and b).

The relationships between a line object O_l and an area object O_a can be found by checking for each edge that is part of the line object how it is related to the area object. This will be expressed by the functions

$$Le[O_l, O_a \mid e_i] = Le[e_i, O_a] \times PartLO[e_i, O_l]$$
$$Ri[O_l, O_a \mid e_i] = Ri[e_i, O_a] \times PartLO[e_i, O_l]$$

For the relationship between a line object O_l and an area object O_a we can write

$$\begin{aligned} B[O_l, O_a \mid e_i] &= Le[O_l, O_a \mid e_i] + Ri[O_l, O_a \mid e_i] \\ &= B[e_i, O_a] \times PartLO[e_i, O_l] \end{aligned}$$

If this function has the value = 2 then the line object runs through the area object at edge e_i if the value = 1 then it is at the border and if it is = 0 then there is no relationship. The relationship between the two objects might be different at different edges.

The Graph Structure of Rasters

Objects might also be represented in a raster geometry. To understand the similarity between such a representation and that of the previous section the cells of a raster will be interpreted as faces. A graph can be linked to a raster according to the generic structure of figure 6; edge$\{n_{i,j}, n_{i+1,j}\}$ is the border between cell$_{i,j-1}$ and cell$_{i,j}$ and edge$\{n_{i,j}, n_{i,j+1}\}$ is the border between cell$_{i-1,j}$ and cell$_{i,j}$. In fact we have:

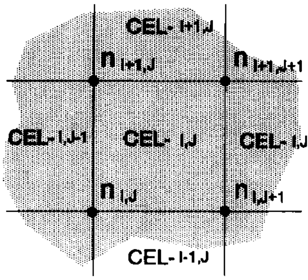


fig. 6: A generic structure for linking a graph to a cell raster.

$Le[\ edge\{n_{i,j}, n_{i+1,j}\},\ cell_{i,j-1}] = 1$ and
$Ri[\ edge\{n_{i,j}, n_{i+1,j}\},\ cell_{i,j}] = 1$

and

$Le[\ edge\{n_{i,j}, n_{i,j+1}\},\ cell_{i,j}] = 1$ and
$Ri[\ edge\{n_{i,j}, n_{i,j+1}\},\ cell_{i-1,j}] = 1$

The only difference between this interpretation of a cell raster and the planar graph with its faces of the previous section is that the cells are rectangular faces so that the relationship between the number of nodes, edges and cells is fixed. Due to this structure each edge has a unique pair of adjacent faces, so that instead of specifying the edges by their begin- and end nodes, they can be specified by the cells at their right hand- and left hand sides, so:

$Edge\{\ n_{i,j}, n_{i+1,j}\} \equiv Edge\{\ cell_{i,j},\ cell_{i,j-1}\}$ and $Edge\{\ n_{i,j}, n_{i,j+1}\} \equiv Edge\{\ cell_{i-1,j},\ cell_{i,j}\}$

With this interpretation cell rasters appear to be special cases of a planar graphs. This conclusion implies that terrain objects can be represented in rasters in the same way as they can be represented in vector maps with a planar graph structure.

Area Objects Represented in a Raster

The geometry of an area object is represented in a raster by one or more adjacent cells. If a cell$_{i,j}$ is part of an area object O_a this will be represented by:

$PartAO[cell_{i,j},\ O_a] = 1$

The relationships between edges an area objects can be found via their relationship with the cells by:

$Le[Edge\{n_{i,j},\ n_{i,j+1}\},\ O_a] \equiv Le[Edge\{cell_{i-1,j},\ cell_{i,j}\},\ O_a] = PartAO[cell_{i,j},\ O_a]$ and
$Le[Edge\{n_{i,j},\ n_{i+1,j}\},\ O_a] \equiv Le[Edge\{cell_{i,j},\ cell_{i,j-1}\},\ O_a] = PartAO[cell_{i,j-1},\ O_a]$

and

$Ri[Edge\{n_{i,j},\ n_{i,j+1}\},\ O_a] \equiv Ri[Edge\{cell_{i-1,j},\ cell_{i,j}\},\ O_a] = PartAO[cell_{i-1,j},\ O_a]$ and
$Ri[Edge\{n_{i,j},\ n_{i+1,j}\},\ O_a] \equiv Ri[Edge\{cell_{i,j},\ cell_{i,j-1}\},\ O_a] = PartAO[cell_{i,j},\ O_a]$

The combination of these two functions gives again

$B[e_*, O_a] = Le[e_*, O_a] + Ri[e_*, O_a]$

If line objects are to be represented in a raster then their geometry should be represented by means of the edges. This will restrict their shape and position to the geometry of the rectangular cells, but the similarity with the structure of the vector approach is then maintained.

The Syntactic Similarity of Vector and Raster Maps

It should be stressed that for the discussion about the similarity of the syntactic structure of raster and vector representations rasters are considered as a grid of cells. In that case the difference between raster maps and vector maps is only due the fact that a raster can be described by a planar graph where the faces have a fixed geometry, i.e. the faces are rectangular so that there is also a fixed relationship between the faces and the edges and nodes. This implies that the position information for the faces or cells can be inferred directly from the cell number if a proper coding has been applied. The cell numbers can be directly related to the numbers of the nodes of the cell boundary and thus a short cut can be made in the computation of the cell location. In a similar way short cuts can be made for establishing the topological (adjacency) relationships between cells. In vector maps the faces do not have a fixed geometry so that the relation between faces and edges and nodes have more degrees of freedom. The identification of the location of the faces, requires that the related boundary nodes are found through the edges. Similarly the topologic relationships between faces should be established through their boundaries. The difference is then only a difference in computational efficiency for the location and the topologic relationships of the geometric elements, because rasters are special cases of planar graphs in this respect.

The expressive power for both geometries is the same however, because the syntactic structure for relating geometric elements to line and area objects is the same. These relationships have for both geometries been based on the functions $PartLO[e_i, O_l]$ and $PartAO[F_f, O_a]$. These functions express the relationships between respectively an edge and a line object and a face and an area object, in a raster map a cell will be substituted for the face . Through the faces the relationships between the edges and the area objects can be established, i.e. it is possible to identify the edges that describe the boundary of the objects. These relationships are for an edge i and an object a represented by the functions $Le[e_i, O_a]$ and $Ri[e_i, O_a]$. The discussions concerning the relationships between objects and geometry in the next chapter will therefore be restricted to these functions.

These statements about the syntactic similarity of the vector and the raster geometry can be extended to other types of tessellations too.

If the interpretation of this paper with a raster as a collection of cells is not strictly maintained the comparison of vector and raster structures might come out slightly different. The discussion in section 1-10 of (Tomlin,1990) is an example of this fact. There the raster elements are considered to be grid points that carry the information about a rectangular zone from which they are the centre points whereas in this paper the raster elements are cells that cover exactly the area they represent. These interpretations are semantically different in the sense that the relation between the elements of the raster data model and the real world are different, resulting in a different interpretation of the geometric relationship between vector and raster representations.

3. SEMANTICS AND UNCERTAINTY

3.1. Uncertainty Aspects of Spatial Objects

The syntax that has been developed in the previous chapters can be considered as a part of a linguistic model for geo-data. The syntax describes how to link object identifiers to information about object classes, thematic attributes and geometric data. With this syntax statements can be formulated about objects and their mutual relationships.

We should see now how the semantics of fuzzy spatial objects can be handled in this syntax, this is important because statements about real world phenomena will always have a certain level of indeterminacy. This is partly due to the fact that the use of data models requires that real world phenomena are described as discrete categories. So we are forced to a discretisation that can not be done with an absolute precision or certainty, there will always be some doubt wether the mapping of the real world onto the discrete categories has been done correctly or adequately. Further more the introduction of this paper explained that data handled in a GIS environment refer to objects that are in fact conceptual entities within some semantical framework defined by the users context of the data. More often then not the users are not very specific about the definition of their objects and object classes, etc. That means that statements about the real world are to be formulated and understood within a certain context (Molenaar,1993). For such statements it is not only important to know how the semantics can be handled by the syntax but the uncertainty related to the statements should considered too. A linguistic model for geo-data should therefore be able the handle the interrelations of the three-tupple: syntax, semantics and uncertainty. This has been illustrated in figure 7; the central part of this figure represents a linguistic model taking care of these interactions, data models and data processing models can then formulated as implementations of this linguistic model in an information systems environment.

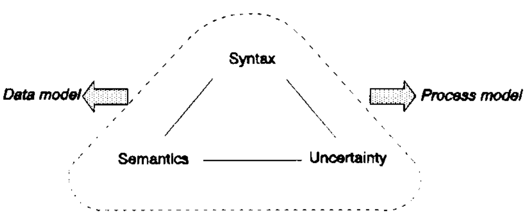


fig. 7:A linguistic model should interrelate syntax, semantics and uncertainty.

Uncertainty is related to statements such as: $x \in S$, i.e. element x belongs to (sub)set S, (see (Molenaar,1993)). Statements of this type can be grouped in three categories related to the information categories defined in the previous chapters:

- S is a subset of the universe $\mathcal{U}_M$, i.e. S is an object class in the sense of section 2.2 and $x \in S$ means that an object x will be assigned to class S,
- S is a subset or a value of an attribute domain, the formula means in that case that a value will be assigned to an attribute of some object,
- S is a region covered by some object, S is then a subset of the point set representing the geometric space of the map , the formula means in that case that some face is part of that region.

The uncertainty of such statements implies the risk that they lead to wrong decisions, with the consequence that inadequate actions will follow. This of course should be avoided, or at least the risk should be brought down to an acceptable level. Therefore we should understand the causes of uncertainty and also we should understand which are the different kinds of uncertainty. The formula "$x \epsilon S$" has three components, uncertainty can be related to each of them (Klir and Folger,1988).

Firstly the definition of a subset S may be fuzzy. No sharp criteria can be formulated to determine whether an element x belongs to S or not. The theory of fuzzy subsets gives mathematical rules for handling this type of uncertainty. Fuzzy subsets are distinct from classical "crisp" subsets in the sense that for crisp subsets the membership function $M[x,S] = 1 \rightarrow x \epsilon S$ (i.e. x belongs to S) or $M[x,S] = 0 \rightarrow x \notin S$ (i.e. x does not belong to S). For fuzzy subsets $0 \leq M[x,S] \leq 1$, hence x may belong a little bit to S. The rules formulated in this theory are mainly of a qualitative nature because it is often difficult to evaluate M[x,S] for fuzzy subsets (Klir and Folger,1988).

Secondly the description of x may be not precise. In GIS x stands for a terrain object for which the geometry and the attribute values should be evaluated. In many cases this will be done through measuring procedures or through the processing of measuring data. Measuring operations introduce in general stochastic components in the observed data. These stochastic components propagate through the processing steps applied to these data. The uncertainty introduced here can then be dealt with mathematically by means of stochastic models. This means that the uncertainty can often be expressed in terms of variances and probabilities.

Thirdly there may be insufficient evidence to assign an element x to a subset S. The problem is now that it is not clear whether a particular element x fulfils the criterion or not. In such situations the subset S might be crisp and the description of the element x might be precise, but still we are not sure whether we should decide $x \epsilon S$ or $x \notin S$. The mathematical rules for handling this type of uncertainty are given by the theory of evidence or the theory of fuzzy measures (Klir and Folger,1988). Sometimes it is possible to evaluate these fuzzy measures, e.g. when they can be expressed as likelihood functions or as Bayes probabilities. But more often this is not the case, so that the rules are again of a qualitative nature.

3.2. Handling the Uncertainty of Spatial Objects

Before the object data are entered into the data base a decision should be made about their description structure. For their geometric description a choice should be made between the raster and the vector structure. We saw that this is mainly a choice between the computational efficiency for geometry of the raster structure verses the geometric flexibility of the vector approach. There are no semantic differences between these two approaches. For the thematic description of the objects the relevant classes with their attribute structure and the attribute domains should be specified.

When the descriptive structure (i.e. the syntax) of the spatial object representation has been defined the object data must be expressed in this structure. The syntax of this paper identified three categories of assignment statements that might involve uncertainty: the assignment of objects to object classes, the assignment of attribute values to objects and the assignment of a spatial description to objects. These three categories will be shortly reviewed here, the emphasis will be on the geometry though.

Fuzzy Object Classes

The intension of the object classes can be expressed by a decision function that tests the objects and then assigns objects to the classes if they pass the test. If an object O is assigned to a class C then M[O, C] = 1 otherwise M[O, C] = 0. If the test does not give definite results then the assignment of object O to class C will be uncertain, that will be expressed by a fuzzy class membership function that does not take definite values 0 or 1. In that case we find $0 \leq M[O, C] \leq 1$. Section 3.1 explained that there could several types of reasons for this uncertainty. The function specifying the classes of O defined in section 2.2.1 should now by modified so that:

$$CLASS(O) = \{C \mid M[O, C] > 0\}$$

The extension of this function might contain several classes even if they are considered to be mutually exclusive. In that case the object can in principle belong to only one class but the uncertain decision function does not give a definite conclusion which one. The object inherits the attribute structures of the classes it belongs to, the attributes should be evaluated. It might very well be that the data required for the evaluation of the attributes contain sufficient information for a definite class assignment in case of mutually exclusive classes.

Fuzzy Attribute Values

The domains of the attributes may contain values or value classes that are not clearly defined. Another possibility is that the values of the domain are clearly defined, but that there is no sufficient reliable measuring procedure to get these values with a high accuracy. The evaluation of an attribute A for an object O should then be expressed by a two-tupple $A[O] = (a, m_u)$, where a is the estimated attribute value and m_u is a measure of the uncertainty that a is a correct value. The uncertainty measure m_u should be of a type related to the type of uncertainty in the sense of section 3.1.

Fuzzy Object Geometry

The geometric description consists of the topology of the objects, their shape and their position. The ***topology*** can be expressed through the relationships between the objects, the faces and the edges. For both the raster and the vector geometry the basic link between area objects and geometry is made through the function PartAO[F, O], in the vector geometry F represents a face and in the raster geometry it represents a raster cell. If the objects O are fuzzy in the sense that their spatial extension is uncertain then this can be expressed by the fact that the function does not only take the value 0 or 1, but that it can take any value $0 \leq PartAO[F, O] \leq 1$. That means for the raster geometry that the resolution, i.e. the cell size, should be chosen so that the value of this function is homogeneous per cell. In the vector geometry the faces should be defined so that the function is homogeneous per face. The fuzzy set of faces related to object O_a is then

$$Face(O_a) = \{ F_j \mid PartAO[F_j, O_a] > 0\}$$

The uncertainty of the topologic relationships between edges and objects can be established through the cells or faces, the uncertainty of the function PartAO[F, O] will propagate to the functions Le[e, O] and Ri[e, O]. This can be done in two steps:

- for each combination of an edge e_i ,a face F_f and an object O_a we find

$$Le[e_i, O_a \mid F_f] = MIN[\, Le[e_i, F_f], PartAO[F_f, O_a]\,]$$
$$Ri[e_i, O_a \mid F_f] = MIN[\, Ri[e_i, F_f], PartAO[F_f, O_a]\,]$$

the functions Le[e, F] and Ri[e, F] however do still have the values 0 or 1 due to the geometric structure of the object description.
The second step is the evaluation of the functions

$$Le[e_i, O_a] = MAX_{Ff}[\, Le[e_i, O_a \mid F_f]\,]$$
$$Ri[e_i, O_a] = MAX_{Ff}[\, Ri[e_i, O_a \mid F_f]\,]$$

If an edge has a face at left hand side or right hand side that has an uncertain relationship to an area object O_a then the uncertainty will be propagated to the functions at the left hand side of these equations. The expressions can be simplified for the raster geometry because of the fixed relationships between cells and edges.

The ***shape and position*** information is mainly contained in the coordinates of the nodes of the object boundaries. Because of the uncertainty of the functions Le[ ] and Ri[ ] the boundary of objects can not be determined in the simple way of section 2.3.1, for fuzzy objects boundaries can only be established for specified certainty levels. Let the fuzzy set of faces related to object O_a with certainty level c be

$$Face(O_a \mid c) = \{\, F_j \mid PartAO[F_j, O_a] \geq c \}$$

With this set we can define the conditional functions

$$\begin{aligned} PartAO[\, F_j, O_a \mid c\,] &= 1 \Rightarrow F_j \in Face(O_a \mid c) \\ &= 0 \Rightarrow F_j \notin Face(O_a \mid c) \end{aligned}$$

and for the relationships between edges, faces and objects

$$Le[e_i, O_a \mid F_j, c] = Le[e_i, F_j] \times PartAO[\, F_j, O_a \mid c]$$
$$Ri[e_i, O_a \mid F_j, c] = Ri[e_i, F_j] \times PartAO[\, F_j, O_a \mid c]$$

These conditional functions are not fuzzy, they take the values = 0 or = 1. Through these functions it can be verified whether an edge is conditionally related to the object by evaluation of the functions

$$Le[e_i, O_a \mid c] = MAX_{Fj}[\, Le[e_i, O_a \mid F_j, c]\,]$$
$$Ri[e_i, O_a \mid c] = MAX_{Fj}[\, Ri[e_i, O_a \mid F_j, c]\,]$$

These are also crisp functions that take the value = 1 if there exist a relationship and = 0 otherwise. With the conditional function

$$B[e_i, O_a \mid c] = Le[e_i, O_a \mid c] + Ri[e_i, O_a \mid c]$$

the boundary of O_a at certainty level c can be found

$$\partial_C O_a = \{ e_i \mid B[e_i, O_a \mid c] = 1 \}$$

The precision of the position and shape of these boundaries is then a function of the accuracy of the coordinates of the nodes. This accuracy depends mainly on the measuring procedure and on the idealisation accuracy of the face boundary, i.e. the accuracy of the identification of the face boundary. In rasters the node positions have been defined by the cell geometry, so that position and shape accuracy of the boundaries at specified certainty levels follows directly from the accuracy of the raster definition.

The ***fuzzy overlap*** between two area objects is the intersection of their two face sets

$$Overlap(O_a, O_b) = Face(O_a) \cap Face(O_b)$$

A fuzzy measure for the fact that the two objects overlap at face f_j is

$$Overl[O_a, O_b \mid F_j] = Min[\ PartAO[F_j, O_a],\ PartAO[F_j, O_b]\ ]$$

A fuzzy measure for the fact that there is some overlap of the two objects can be found by

$$Overl[O_a, O_b] = Max_{Fj} [\ Overl[O_a, O_b \mid F_j]$$

Fuzzy relationships between a line and an area object should be found through their common edges

$$Le[O_l, O_a \mid e_i] = MIN[\ Le[e_i, O_a],\ PartLO[e_i, O_l]\ ]$$
$$Ri[O_l, O_a \mid e_i] = MIN[\ Ri[e_i, O_a],\ PartLO[e_i, O_l]\ ]$$

The relationships between the edge and the area object are fuzzy. If the relationship between edge and line object is crisp then the function PartLO[ ] takes the values = 0 or = 1, if this relationship is also fuzzy the it can take any value between 0 and 1. If O_l intersects O_a at e_i then it should have O_a at both sides, a fuzzy measure for this intersection is given by $MIN[\ Le[O_l, O_a \mid e_i],\ Ri[O_l, O_a \mid e_i]\ ]$. A fuzzy measure for the existence of some intersection between O_l and O_a given by $MAX_{ei} [\ MIN[\ Le[O_l, O_a \mid e_i],\ Ri[O_l, O_a \mid e_i]\ ]\ ]$

The relationships between edges, faces and objects explained in this section can be used to derive the fuzzy variants of more topologic relationships between objects such as adjacency, touch, containment etc.

4 CONCLUSION

A clear distinction should be made between the models that describe how scientists from different disciplines perceive the world and the representation of these models in an information systems environment. The different descriptive models of the various disciplines should be mapped onto discrete data sets with their mutual relations and related data processing models if they are to be implemented in an information system. The previous sections presented some syntactic structures for geo-data models with the emphasis was on the structure of state descriptions of objects.

There are several possibilities for the geometric description of spatial objects, these are given by the different types of tessellations that are available. The most common tessellations for GIS applications are given by the vector structure (or the general form of cell complexes) and rasters. It seems that these approaches do have basically the same syntactic structure for the representation of spatial objects and this conclusion can in principle be extended to the other tessellations too. The choice between the different tessellations is in fact a choice between the flexibility of geometric description versus computational efficiency for position and topologic relationships. The expressive power of these geometric models is the same. This implies that there is syntactically also no difference between these geometries when fuzzy objects are to be represented. The links between these objects and the geometric elements can be expressed as fuzzy relations, through these fuzzy topologic relationships and fuzzy position data can be derived. The results of the previous sections seem to indicate that fuzzy spatial object data can to a large extend be handled by not too dramatic modifications of the present spatial data models.

LITERATURE

Carnap, R. (1956) *Meaning and necessity, a study in semantics and modal logic.* (The Univ. of Chicago Press, scnd edit. 56), Midway Reprint Edition 1988.

Chen, P.P.S. (1976) The Entity-Relationship Model - Toward a Unified View of Data. *ACM Transaction on Database Systems*, vol. 1, no. 1, pp9-36.

Frank, A.U. and W. Kuhn (1986) Cell Graphs: A Provable Correct Method for the Storage of Geometry. *Proceedings of the 2nd International Symposium on Spatial Data Handling*, Seatle.

Gersting, J. L. (1993) *Mathematical structures for computer science.* New York: Computer Science Press, Third edition.

Klir, G.J. and Folger, T.A. (1988) *Fuzzy sets, uncertainty, and information.* Englewood Cliffs (N.J.): Prentice-Hall.

Molenaar, M. (1991a) Terrain objects, data structures and query spaces, in Schilcher, M. (ed.) *Geo-Informatik*, pp53-70, München: Siemens-Nixdorf Informationssysteme A.G..

Molenaar, M. (1991b) Formal data structures and query spaces, in Günther, O., e.a., (eds) *Konzeption und Einsatz von Umweltinformationssytemen,* pp340-363, Berlin: Springer-Verlag.

Molenaar, M. (1993) Object Hierarchies and Uncertainty in GIS or why is Standardisation so Difficult. *Geo-Informations-Systeme*, vol.6, no.4, pp22-28.

Tomlin, C.D. (1990) *Geographic Information Systems and Cartographic Modelling.* Englewood Cliffs (N.J.): Prentice Hall.

Ullman, J.D. (1982) *Principles of Data base systems.* Rockvill: Computer Science Press Inc.

Multi-scale Modelling

CONTEXTUAL TRANSFORMATIONS AND GENERALIZATION OF REMOTELY SENSED IMAGERY FOR MAP GENERATION

Dianne E. Richardson
Canada Centre for Remote Sensing
Natural Resources Canada
588 Booth Street
Ottawa, Canada
K1A 0Y7
internet: drichard@ccrs.emr.ca
tel: 613-947-1252
fax:613-947-1383

ABSTRACT. Contemporary models in spatial and ancillary generalization have focussed on the development of rule bases which by definition are limited in scope. This paper presents an object oriented model to handle the automatic classification, coding, and database building. It uses techniques based on underlying data structures that can be manipulated to render automatic and continuous generalization to meet contextual requirements such as topographic and aeronautical chart mapping as well as thematic mapping at any smaller scale. This approach uses object oriented techniques for data abstraction and can provide results that can respond to a diversity in requirements. The model does not restrict the user to a pre-defined and finite rule-base.

1. INTRODUCTION

In spatial and attribute generalization, data abstraction can occur differently at different scales of representation. Even at a single scale, data abstraction can vary considerably since applications in using the data change from user to user. As such, a data model environment should allow a maximum of flexibility in data manipulations while at the same time providing a stable environment. Data modelling is therefore an important aspect in creating any prototype for automatic generalization. The data model design should allow enough flexibility to permit a class or a portion of a class of abstracted data to be represented in different ways, and conversely, to allow different classes of abstracted data to be viewed as a single class of data when required by scale or context. For example, when generalizing remotely sensed imagery for 1:250 000 topographic mapping, the data model should provide automatic transformations of smaller and less significant river linked lakes to be viewed as rivers. The same applies for the topographic representation of built up areas comprised at a more detailed level of residential, commercial, manufacturing/industrial and open spaces.

Ideally, the data model would support all interpretations of geographic phenomena according to the application or generalization need. Since this is unlikely, it is therefore important that the data model capture the appropriate amount of meaning as related to the desired use of the data, and in addition, be designed in such a way as to accommodate unanticipated requirements.

This paper examines two aspects of a generalization prototype. The first aspect is a conceptual data model for generalization that presents solutions for quantitatively reducing data density according to a number of contextual issues. The second set of aspects considered relate to object oriented data structures and classifications, and their manipulation in generalization and specialization hierarchies.

2. OBJECTIVES

The first objective is to determine and develop ways data can be automatically structured, coded and classified in hierarchies. These techniques rely on topological relationships as well as geometric configuration and are necessary in generalizing hydrographic and transportation networks. They also provide a solution for automatic database building.

The second objective is to present an approach to spatial and thematic generalization which utilizes and exploits the derived database using object oriented constructs. Since generalization activities require flexibility, the data model design and subsequent lower level data structures should be built accordingly. One way in which this can be achieved is through a number of options in manipulations of classification and aggregation hierarchies. These structures support object derivations, as well as the manipulation of subclass, class and superclass levels of geometric and thematic data. Accordingly, new objects can be built, and data maintained in the database can be abstracted to render different densities as well as provide new distributions according to contextual requirements.

A third objective is to create a means for controlling the amount of generalization. Here the objective is to provide an environment that moves away from interval generalization based on discrete scales and to develop a technique that allows continuous generalization. Thus, a module is provided that allows a user to continuously decrease entity, object, subclass, class and superclass densities.

3. METHODOLOGY

Four areas of knowledge are important for the model development. The first area concerns the users need. In addition to factual information, a number of assumptions can be made about anticipated diversity expected from different users. The second area of knowledge is in spatial and thematic generalization. Although many activities fall under the term cartographic generalization, the four primary activities that lead to an abstracted representation are classification, selection, simplification, and symbolization. Of particular interest are classification and selection activities, since they deal with geographic content, while other activities such as simplification, displacement, and symbolization handle representation issues. The third area is the data model environment which must present and support suitable structures for data manipulation. Topological relationships should be defined as well as the design of any additional information required for aspects such as object collapse, or coalescence. The fourth area to be explored concerns the logical data structures which are built according to the conceptual data model and according to requirements specified by the users.

The four areas of knowledge are translated to three levels of architecture which are referred to as the external level, conceptual level, and internal level. The external level reflects the user's needs. At this level, mapping requirements are assessed along with requirements for changes in context of a map or representation. The next level addressed is the conceptual level. This is the model view which should reflect the external level user's requirements. At this level, two models are designed. One is the context transformation model which provides techniques for data abstractions and continuous generalization, while the other conceptual model defines the data model environment that handles the lower level transformations. At the internal level, geometric and ancillary data in raster and/or vector format are implemented and tested. See figure 1.

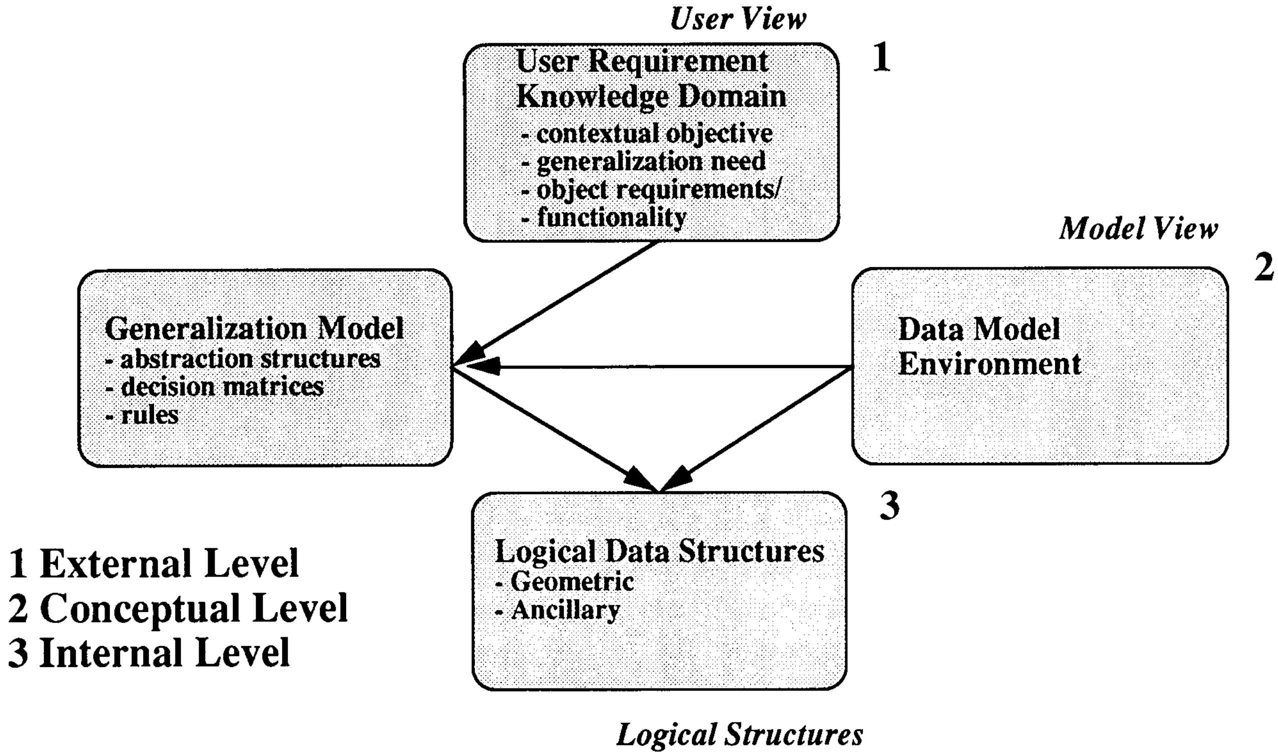
3.1 External Level - Users Needs

An analysis of the external level or in other words, the user assessments indicated that the range of generalization possibilities according to a specified context are legion. As a result, only two aspects are considered, those of spatial analysis or modelling, and map design. Before briefly exploring these considerations, a number of a-priori assumptions about user requirements should be outlined. For example, given the nature of users, it is assumed that multiple abstractions would be required from a single database. Additionally, multiple interpretations from the same data are considered necessary. An example of this is in the classification generalization of urban areas which may consist of housing, manufacturing, and commercial zones. A new abstraction may require their generalization to built up areas. Thus it can be considered that at different levels of representation, aggregated data sets with a subsequent new interpretation of ancillary data would be required in a generalization system.

Four basic external user needs have been considered and consist of a) thematic needs, b) generalization need depending on theme, c) object requirements, and d) object functions. The thematic need depends on the type of mapping. For example topographic requirements are considerably different from aeronautical charts. In the latter case some features are often aggregated on the map since that is how they are perceived by pilots flying at 10 000 feet or higher. Accordingly, rules can be derived for the particular theme under study. Object requirements and functionalities have been defined for a number of different types of map environments such as topographic, aeronautical, and thematic. These requirements are calculated based on scale to provide a guide for decreased object densities. [See Richardson, 1993, 1994].

3.2 Conceptual Level - Generalization Model

Three Levels of Architecture



User View

User Requirement
Knowledge Domain
- contextual objective
- generalization need
- object requirements/
- functionality

1

Model View

Data Model
Environment

2

Generalization Model
- abstraction structures
- decision matrices
- rules

Logical Data Structures
- Geometric
- Ancillary

3

Logical Structures

1 External Level
2 Conceptual Level
3 Internal Level

Figure 1

The conceptual level refers to the model environment. In this case, two models are involved, one for generalization, and the other for the database environment. Both conceptual models must be designed to comply with requirements at the external level. These two models are addressed individually and are as follows.

Data Model:

A number of a-priori assumptions should be made about the data model before its design and implementation in a database environment; accordingly, the data model should a) allow flexibility in using the database, b) allow modelling of locational and ancillary data in an integrated way but also independently, c) permit modelling of thematic data from one level of measurement to another like transforming interval or ratio data to ordinal or nominal, and d) provide the capability to geometrically transform different levels of thematic abstraction to different scales of representation.

In addition to assumptions, a number of criteria need to be met when designing the conceptual data model. For example the conceptual data model should be translatable to logical data structures that allow multiple interpretations. Once implemented, this implies different levels of abstractions of spatial and ancillary data should be possible at different scales of representations and according to the application context. To render these facilities, generalization and specialization hierarchies should be developed for spatial and ancillary data. The model selected to support the development of hierarchical structures is a topological data model referred to as the formal data structure for single valued vector maps. [See Molenaar, 1990, 1991].

This model handles both geometric and thematic aspects of geo-information using elementary data types like points, lines, and areas, and sets of geometric links among data types and objects to provide a 'feature-oriented' data structure. This environment facilitates the analysis of topological relationships among geometric elements and objects and the construction of composite objects.

Generalization Model:

Since the nature of generalization is contextually sensitive, it implies the generalization model must provide techniques with which a user can contextually alter data. This concept can be termed context transformations. Context refers to the whole situation, background, or environment relevant to a particular event. The term transformation refers to the alteration of spatial and thematic data, or rather their change in form or outward appearance, to meet specific contextual considerations. Thus, the term context transformation means a change in context which evolves as a result of the mapping requirement or situation as well as the user's perspective, and, the physical transformation of the data implies a change in data structures as a result of the contextual emphasis.

Context transformations have been tested using topographic, aeronautical chart, and thematic mapping requirements. Additional contexts have been included which reflect spatial analysis and map design requirements.

The first step however, is to calculate the amount of data that should be abstracted from the database. For this, three options are available, Töpfers formulas, the Necessity Factor, and Percentage values. Briefly, these are as follows.

Töpfer: This method calculates the number of features to retain using Töpfers formula. [See Töpfer F., and Pillewizer W, 1966]. This is a default method when 'topographic' generalization is required. Three variations on the formula are available, the basic formula, the formula for unexaggerated representation, and the formula for exaggerated representation. When Töpfer's formula is applied, the same percentage of objects are removed from each object class.

Necessity Factor: The necessity factor (see Richardson, 1989, 1991, 1993) determines the number of objects to select by examining the map object class, the source scale, the destination scale, and the thematic context for the generalization. This is the default method if the purpose of the generalization is thematic. The reduction in data density derived by the necessity factor can be attenuated or amplified to increase or decrease data reduction respectively.

Percentage: In this method, the user specifies the percentage of features to retain. This can be set once for all classes and subclasses, or the percentage of features to retain can be individually adjusted for each specific classification.

Implementation: A prototype referred to as gensystem generalizes map features based on object oriented techniques, using superclass-class and subclass structures. Hydrographic datasets have been tested using these processes.

In the classification hierarchy, classes are mutually exclusive and individual map objects belong to only one class. However, objects of a class can be described in different ways by means of subclasses in which objects of a subclass also belong to their parent class. For example, in a collection of subclasses and classes, at the top of the classification hierarchy is a single superclass. Each class represents objects that have some characteristic in common. The subclass occurs when all the members of a class are also members of some other class. Thus, the classification hierarchy is divided into three levels: the superclass level, the class level, and the subclass level.

The superclass level is the top level in the classification hierarchy. The superclass is the parent class of the classes and subclasses and corresponds to the most generalized representation of the features.

Special considerations are required for handling hydrological objects. The hydrology superclass must be specified with correct directionality, i.e. river segments must topologically follow the direction of flow. Before hydrographic objects can be generalized, stream orders must be calculated, coded and loaded into the database. This is an automatic process based on topological and geometric relationships. Three different stream orders are calculated since they are relevant for different types of contextual generalization. The three used are Horton, Shreve, and Strahler.

Following the determination of how much data should appear, a strategy based on hierarchies should be available to allow the automatic abstraction of objects from the database. This abstraction must meet the specified context with the subset of data required at the smaller scale. Thus, once a knowledge-base provides guidance on how much data should be shown at a reduced scale, techniques should be provided to automatically provide the correct composition.

The classification and aggregation hierarchies can be used to automatically preserve the significant information and remove the random or unimportant. When used in a knowledge-base, frequent iterations are eliminated and the new representation and accompanying database are provided.

4. HYDROGRAPHY ABSTRACTION

In a hydrography superclass, the abstraction activities rely on the classification structure of rivers and on their topological relationships with lakes. The use of classification and aggregation structures support automatic elimination, lake/river replacement, and as well, allow automatic elimination of braided stream segments and delta segments, (Richardson, 1994) To do so, hydrographic data is stored in the form of arc, node topology at the logical data structure level. The arcs form a one-to-many relationship with named rivers. Three classification structures have been used for testing; the Horton, Strahler, and Shreve classification schemes.

Rivers can be abstracted at the class level for any lower level of representation using any of the classifications. The three classifications used result in significantly different representations. The Horton class level abstraction provides a representation of a drainage network that more closely resembles a conventionally derived cartographic product. This is as a result of Horton's classification structure in which a higher weighting is given to the major river network.

The data abstraction process responds to values established using Töpfer, the Necessity Factor or a Percentage value as follows:

$$\forall\ C_k \subset SC \text{ and } I(C_k) > I(C_k + 1) \tag{1}$$

$$\forall\ A_p \in C \text{ and } A_p(e_i) > A_p(e_i + 1) \tag{2}$$

select e_i for $i <= n$ where $\frac{n}{i_{max}} = \%C$

where $\%C = x$

$\forall$ = for each and every

C_k = class k

$\%C$ = percent value specifying abstraction level

SC = superclass

A_p = value of attribute A_p of the object e_i

x = amount of abstraction specified by Töpfer, the Necessity Factor, or a Percentage value selected by the user.

5. SUPERCLASS ABSTRACTION

The superclass abstraction in hydrography selects and eliminates rivers according to either Strahler, Shreve, or Horton but applies the same percent reduction to both rivers and lakes. This allows users to abstract at the superclass level rather than class level. This concept is important when decreasing very detailed databases where classes such as bogs, fens, marshes, and swamps may be contained. Here the use of superclass abstraction reduces the number of iterations that would be required of the user and provides the `superclass' abstraction as determined by a reduction value. The superclass abstraction takes the following form.

$$\forall e_i \in SC \text{ and } A_p\ (e_i) > A_p\ (e_i + 1) \tag{3}$$

select e_i for $i < = n$ where $\frac{n}{i_{max}} = \%SC$

6. SUMMARY

Classification and aggregation hierarchies rely on a classification structure being established in the data under investigation. In hydrography, automatic classification of stream orders using Shreve, Strahler, and Horton is relatively straight forward. The data model supports the data classification by using topology and geometry. In this way, stream orders are automatically coded, braided streams and deltas are handled, and the database is loaded for the user. No preprocessing other than standard topology creation is necessary. Once the database has been built by the system, the values defined by Töpfer, the Necessity Factor, or the percentage values specifies how much data to abstract. This is done according to the knowledge-base for topographic, aeronautical or thematic mapping requirements. The Töpfer process is used generally for the topographic application as well as aeronautical chart, and the Necessity Factor can be used for the thematic representations. Additionally, these measures can support large scale changes, such that Töpfer's solutions are applied first and then followed by the Necessity Factor application. In all cases, however, the composition of the abstracted data is achieved using the hierarchical structures. In this way, the user specifies the derivation of a topographic map, selects the measure for reducing the density, and leaves the system to calculate all aspects. The calculations include classifying the data, building the database, ordering the database, abstracting the correct composition given the classifications used, and determining feature collapse requirements.

These techniques are also applicable to transportation networks. In these cases, the definition of the classification is based on different parameters such as minimum path, optimal path, sub-optimal paths, and accessibility analysis.

REFERENCES

Horton, R.E. 1945. Erosional Development of Streams and their Drainage Basins. *Bulletin of Geological Society of America*, Vol. 56, pp 275-370.

Molenaar, M. 1989. Single Valued Vector Maps; A concept in Geo-Information Systems. *GeoInformationssysteme*, 2(1) pp 18-26.

Molenaar, M. 1990. A Formal Data Structure for Three Dimensional Vector Maps. *Proceedings of the 4th International Symposium on Spatial Data*. Vol. 2, Zurich, Switzerland. pp 830-843.

Molenaar, M. 1991. Object Hierarchies and Uncertainty in GIS or Why is Standardisation so Difficult. in *Kadasterin Perspectief*.J.H.M. Kockelkoren et al, editors. Kadaster, Apeldoorn, The Netherlands.

Molenaar, M., and L.L.F. Janssen, 1992. Integrated Processing of Remotely Sensed and Geographic Data for Land Inventory Purposes. *Proceedings, ISPRS Conference XVII*, August 2-14, 1992, Washington, D.C.

Molenaar, M., and D. Richardson, 1993. Linking Aggregation Levels for Multiple Abstractions in GIS, *Proceedings, International Workshop on Global GIS, ISPRS*, August 24-25, Tokyo, Japan.

Nyerges, T. L. 1991. Representing Geographical Meaning; in *Map Generalization: Making Rules for Knowledge Representation*. B.Buttenfield and R. McMaster, editors, Longman House Essex, United Kingdom.

Richardson, D. 1989. Rule Based Generalization for Base Map Production. *Proceedings National Conference on GIS; Challenge for the 1990's*. Ottawa, Canada. pp 718-739.

Richardson, D. 1991. Conceptual Model for Generalization. *15th Conference of the International Cartographic Association*. Bournemouth, England, October 1991.

Richardson, D. 1993. *Automated Spatial and Thematic Generalization Using a Context Transformation Model*. R&B Publications, Ottawa, Canada, ISBN 0-9697043-0-5.

Richardson, D. 1994. Generalization of Spatial and Thematic Data Using Inheritance and Classification and Aggregation Hierarchies. Spatial Data Handling Conference '94. Edinburgh, Scotland, September 5-9, Great Britain, paper accepted.

Richardson, D. and J.-C. Muller. 1991. Towards Rule-Based Selection of Spatial Objects for Small Scale Map Generalization; in *Map Generalization: Making Rules for Knowledge Representation*. B. Buttenfield and R. McMaster, editors, Longman House Essex, United Kingdom.

Rumbaugh, J., and M. Blaha, W. Premerlani, F. Eddy, W. Lorensen. 1991. *Object-Oriented Modelling and Design*, Prentice Hall Englewood Cliffs, New Jersey, ISBN 0-13-629841-9.

Smith, J.M. and D.C.P. Smith. 1989. Database Abstractions: Aggregation and Generalization. in *Readings in Artificial Intelligence and Databases*. J. Mylopoulos and M. Brodie, editors, Morgan Kaufmann Publishers, Inc. San Mateo, California.

Strahler, A.N. 1964. Quantitative Geomorphology of Drainage Basins and Channel Networks. *Handbook of Applied Hydrology*, Ven Te Chow, editor.

Thompson, P. J. 1989. *Data With Semantics: Data Models and Data Management*. Van Nostrand Reinhold, New York.

Töpfer, F. and W. Pillewizer, 1966. The Principles of Selection, a Means of Cartographic Generalization. *The Cartographic Journal*, 3(1):10-16.

MULTIRESOLUTION TOPOLOGICAL MAPS

M. BERTOLOTTO, L. DE FLORIANI
Dipartimento di Informatica e Scienze dell'Informazione
Università di Genova
Viale Benedetto XV, 3, 16132 Genova, ITALY

E. PUPPO
Istituto per la Matematica Applicata
Consiglio Nazionale delle Ricerche
Via De Marini, 6 (Torre di Francia), 16149 Genova, ITALY

ABSTRACT. A formal framework is defined, for representing plane maps at different levels of resolution, that is based on a hierarchy of topological models described by a tree. Spatial objects are explicitly represented in the context of a topological model at each level of resolution. Multiresolution is based on the refinement of the representation of a spatial entity into a map at higher precision and detail. Different representations of an entity at different levels of resolution are formally related through the concept of homotopy.

1. INTRODUCTION

The representation of spatial data at different scale/resolution in the context of a unified model is a topic of relevant interest in spatial information theory. Indeed, multiresolution modeling offers interesting capabilities for spatial representation and reasoning: from support to map generalization and automated cartography, to efficient browsing over large GISs, to structured solutions in wayfinding and planning (through either bottom-up or top-down techniques on a hierarchy of spatial data).

Current GISs do not offer much in multiresolution data handling: apart from some hierarchical capabilities in raster modeling, which are essentially based on structures and tools inherited from image processing, there is an almost total lack of features for handling and relating spatial data at different resolutions. In order to support GISs of future generations, it seems worthwhile to pursue the definition of a formal framework for multiresolution representation of spatial entities based on a topological model, that offers explicit description of spatial objects, and efficient encoding/retrieval of spatial relations.

In (De Floriani and Puppo 1992) we have proposed a model, called HTIN, for the multiresolution representation of terrain models, that is based on a hierarchy of Triangulated Irregular Networks (TINs). The idea underlying the HTIN is to represent a terrain through a TIN at a given level of precision, and to recursively refine the representation inside each triangular region into another TIN, that represents such region at a higher level of precision. The resulting structure is a tree of TINs, where the root corresponds to a description of the whole terrain at the coarsest resolution, while each internal node has a triangular domain and corresponds to the refinement of the surface defined by a triangular patch in its parent node. In this paper, we do not address the representation of terrains. We rather exploit and extend some principles of the hierarchical organization underlying the above mentioned model, to define a multiresolution model for plane maps.

If we consider the triangulation induced on the plane domain by each TIN, the HTIN can indeed be regarded as a hierarchical nested subdivision of the domain into simplicial maps. The main characteristic of such model is that the internal structure of a region, represented by a triangle at a given resolution, can be detailed into a map of the same triangular area at a higher resolution.

Although some authors have proposed models for plane maps that are based on simplicial complexes (Frank and Kuhn 1986, Egenhofer, Frank and Jackson 1989, Worboys 1992), more general models exist, that allow representing maps formed by regions of arbitrary polygonal shape, and containing isolated points and lineal features (Molenaar 1989, Pigot 1992, De Floriani, Marzano and Puppo 1993). Besides, the main difficulty in directly extending our hierarchical structure to multiresolution maps is the strong geometric constraint that imposes the coincidence between the boundary of a region and the boundary of the map refining its internal structure. Indeed, it is well known that polygonization induces horizontal error on lines, hence on the boundaries of regions: such error will be normally higher as the resolution of the map decreases. Thus, when refining the map of a region at a higher resolution, the polygon bounding such region will slightly change into a different polygon, normally formed by a higher number of edges, and characterized by a lower horizontal error.

The subject of this paper is the definition of a hierarchical model for maps, that is described by a tree of topological models. Our basic topological model is the *Plane Euclidean Graph* (PEG), that we introduced in (De Floriani, Marzano and Puppo 1993) for representing maps containing generic polygonal regions with holes, isolated points and lineal features. Similarly to the HTIN, each node in the tree is a PEG corresponding to the refined map of a polygonal region in its parent node, also containing the refinements of its lineal features. We relax the geometric constraint – that in the HTIN imposes coincidence of domains between a region and its refinement – into a topological constraint, that is based on the notion of homotopy, and imposes a consistency of the overall shape of a region, of its holes, its adjacencies in the map, and its features, throughout the whole hierarchy.

2. A TOPOLOGICAL MODEL FOR MAPS

In (De Floriani, Marzano and Puppo 1993) we have introduced a topological model for maps, that is briefly described in the following, and that constitutes the basic model for defining our hierarchical maps.

A plane map is composed of geometric entities of three classes, namely *points*, *lines*, and *regions*, embedded in the Euclidean plane. Elements of each class have a well-defined structural and geometric characterization, that allows a complete and concise description of each entity. In particular, we consider simple open lines, and regions, possibly with holes, bounded by a set of closed and simple chains of lines: a chain defines the *outer boundary* of a region; other chains may define *inner boundaries*, separating a region from other regions completely contained into it. By Jordan theorem, a region is defined as the portion of plane inside the outer boundary and outside all inner boundaries.

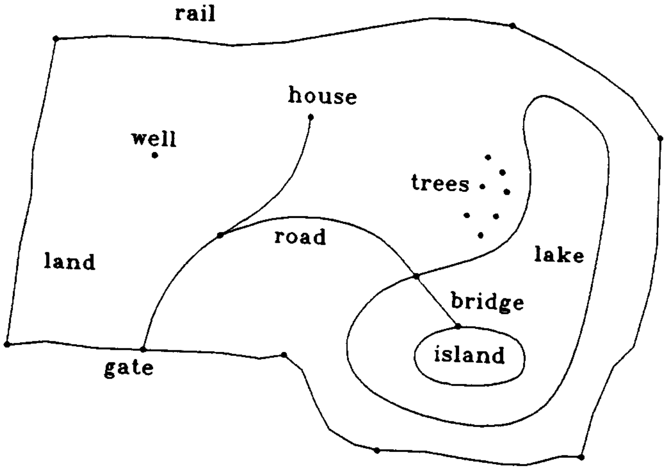


Figure 1: An example of map.

We define a *map* as a triple $M = (P, L, R)$, where P is a set of points, L is a set of lines, and R is a set of regions. For the consistency of the map, L shall contain at least all lines defining the boundary of entities in R, and, similarly, P shall contain at least all endpoints of entities in L. Anyway, a map M does not necessarily contain *only* points and lines that are part of the boundary of some other lines and regions of the map, respectively: isolated points and lines can be present, that denote *features* of the map. Figure 1 shows a simple example of map containing regions that are not simply connected, as well as point and lineal features.

In the context of a single map, we do not allow general intersection between spatial entities: lines are broken into chains by inserting points in the map wherever a line intersects another line. This constraint ensures the planarity of the graph defined by points and lines of the map, thus guaranteeing that topological relations between spatial entities are well defined. Such map model is conceptually analogous to the *single valued vector map* proposed in (Molenaar 1989).

In our work we consider approximate representations of generic simple lines through polygonal chains: the geometry of a polygonal chain is represented by a sequence of points in the plane. As a consequence, also regions are approximated by polygonal regions.

A map is represented by a *Plane Euclidean Graph (PEG)*, that is defined by a graph $G = (V, E, F)$, embedded in the Euclidean plane, where:

- V, called the set of *vertices*, is a set of points in the plane;

- E, called the set of *edges*, is a set of polygonal chains having their endpoints in V, and such that any two edges of E never cross (i.e., they never intersect, except at their endpoints). A *polygonal chain* of E is defined as a sequence of points $e = (v, pl, \dots, p_k, w)$, where $v, w \in V$ and $p_1, \dots, p_k \notin V$. Points $p_1, \dots, p_k$ are called *joints* of e.

- F, called the set of *faces*, is a set of maximal regions f bounded by chains of E,

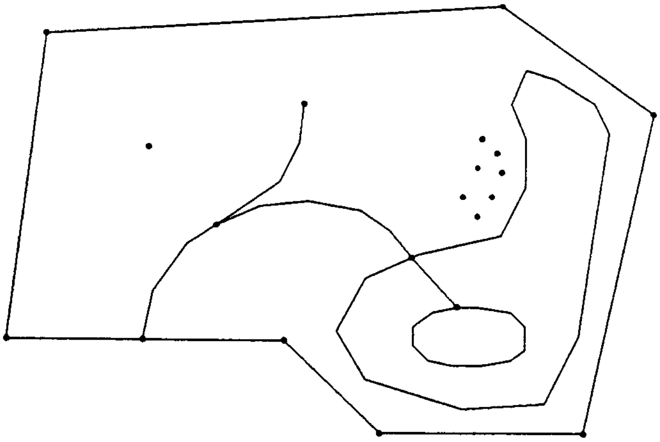


Figure 2: A PEG describing the map of Figure 1.

such that for every two points P and Q inside f, there exists a curve on the plane that joins P to Q without intersecting any polygonal chain of E.

Note the different role that vertices and joints play in the model: vertices affect the topology of the map and should correspond to either branch points or points features, while joints have no topological meaning, and are introduced only to specify the geometry of polygonal edges.

Given a map $M = (P, L, R)$, defined as above, we represent M through a PEG $G_M = (V, E, F)$ where $V \equiv P$, $E \approx L$, and $F \approx R$, where symbol $\approx$ is used to mean that each element in E (respectively, F) is an approximate representation of an entity in L (respectively, R); this notation is intended to make a distinction between spatial entities and their representations embedded in the topological model. Figure 2 depicts a possible PEG encoding the map of Figure 1. Note that the vertices of the PEG are represented as bold dots, while other endpoints of straight-line segments are joints of edges.

Entities in a PEG G are topologically related, like the spatial entities they represent, through adjacency and incidence relationships. Moreover, entities representing point and lineal features of the map are related to edges and faces of the model through containment relations. Spatial relationships between entities in a maps, and their corresponding relations in a PEG, have been formally described in (De Floriani, Marzano and Puppo 1993), where spatial queries based on such relations have been extensively discussed.

3. REFINING LINES AND REGIONS

An edge e in a PEG G is the approximated polygonal representation of a simple line l. The relation between e and l, and the precision of the approximation, can be formalized through concepts rooted in algebraic topology. Our formalization relies on an idea reminiscent of the well-known concept of *band*, introduced in (Peucker 1976).

The *convolution* of a simple line s with a closed disc d of radius ε is the region $R_{s,\varepsilon}$

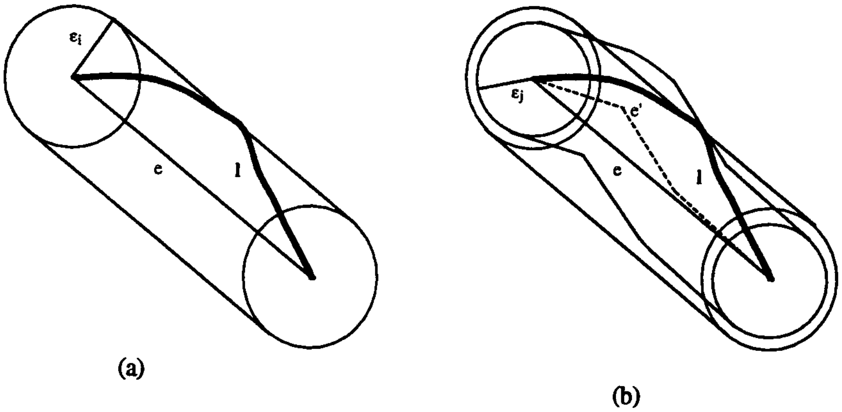


Figure 3: (a) A line l and a polygonal edge e approximating l at precision ε_i; (b) a refinement of e at precision ε_j.

covered by d when sweeping its center along s. For the sake of simplicity, we assume that ε is always small enough to make $R_{s,\varepsilon}$ a simply connected region having s as skeleton.

The set of simple lines that are contained in $R_{s,\varepsilon}$, have the same endpoints of s, and can be continuously deformed to s without leaving $R_{s,\varepsilon}$ defines a *homotopy* class, that we call the ε-homotopy of s.

Given a polygonal edge e as above, we say that e approximates a line l at precision ε_i if and only if ε_i is the smallest among values ε such that l is in the ε-homotopy of e (see Figure 3a). The *refinement* of e at a precision $\varepsilon_j < \varepsilon_i$ will be a simple polygonal chain e' in the ε_i-homotopy of e, and such that l is in the ε_j-homotopy of e' (see Figure 3b).

A simply connected polygonal face f in a PEG G is the approximated polygonal representation of a simply connected region r. The boundary of f is composed of a closed polygonal chain c, formed by a sequence of polygonal edges of G, i.e., $c = (e_0, \ldots, e_k)$, that approximates the boundary b of r. Each edge e_h in c approximates a line l_h of b. If ε_i is the largest error made in approximating l_h by e_h, for each h, then we say that f approximates r with precision ε_i. In other words, the boundary of r will be contained in the ring-shaped region R_{c,ε_i}, that indeed defines the ε_i-homotopy of (closed line) c. The set of simply connected regions whose boundary is in the ε_i-homotopy of c will be called the *ε_i-homotopy* of f. Figure 4a depicts the ε_i-homotopy of a polygonal region.

The largest among such regions, bounded by the outer boundary of R_{c,ε_i}, will be called the *ε_i-fattened region* of f (see Figure 4b); the smallest among such regions, bounded by the inner boundary of R_{c,ε_i}, will be called the *ε_i-kernel* of f (see Figure 4c).

The *refinement* of f at a precision $\varepsilon_j < \varepsilon_i$ will be a polygonal region f' in the ε_i-homotopy of f, whose boundary is formed by a sequence of refinements of edges $e_0, \ldots, e_k$, and such that region r represented by f is in the ε_j-homotopy of f'.

The above definitions are easily generalized to multiply connected regions by considering both approximations of the outer and inner boundaries. For simplicity, we will assume that ε is small enough that the ε-homotopy of different boundaries of a region will not intersect. Indeed, in case ε is too large and such regions intersect, they cannot be treated independently, and the refinement process becomes difficult. Formal definitions for regions

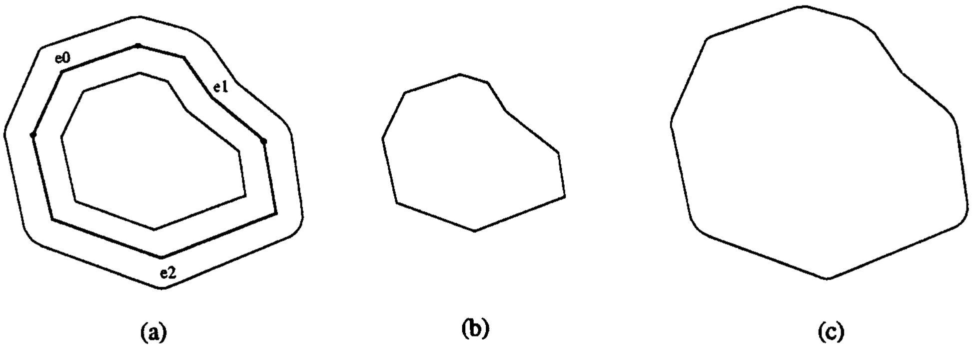


Figure 4: A polygonal region f and (a) its ε_i-homotopy; (b) its ε_i-kernel; (c) its ε_i-fattened region.

with holes are straightforward and are omitted here for brevity.

4. THE HIERARCHICAL MODEL

A *Hierarchical Plane Euclidean Graph (HPEG)* is defined by recursively applying the concept of refinement, introduced in the previous Section, to regions of a PEG. Intuitively, given a PEG G, we can consider each face f of G at precision ε_i as an individual entity. We can refine f at a higher resolution into a PEG G', whose domain is homotopic to f, by refining the boundary of f, and adding new vertices and edges inside the refined domain. In principle, each face of a given PEG is refined independently. The recursive application of the refinement process results in a tree of PEGs.

Note that the idea of refining a face f into a PEG G' involves two concepts:

1. *higher precision*: the boundary of f and the features contained into it are represented in G' with smaller horizontal error;
2. *higher detail*: the region represented by f is divided into subregions, and other features may appear.

While it is straightforward to relate the increasing precision with the concept of refinement introduced in the previous Section, there is no immediate relationship between the error in the representation of a map, and the fact that a given entity appears in it or not. Indeed, in cartography, the relation between the relevance of an entity and the scale of the map can often depend on the size of the entity, but it might also depend on semantic information associated with the map. This is one of the main reasons that make the long-standing problem of map generalization an ill-posed problem. For the sake of simplicity, here we assume that for each entity in the map to be represented, the minimum resolution (i.e., the largest error in representing data) at which the entity appears in a map is known. Under such an assumption, a decreasing sequence of error thresholds, called the *levels of resolution*, is sufficient to control both precision and detail in the definition and construction of our model.

More formally, let $\tilde{\varepsilon} = (\varepsilon_0 > \ldots > \varepsilon_k)$ be a sequence of levels of resolution. An HPEG is described by a tree $\mathcal{H} = (\mathcal{G}, \mathcal{E})$, where $\mathcal{G} = \{G_0, \ldots, G_h\}$ is a family of PEGs, and $\mathcal{E} = \{(G_i, G_j) \text{ for some } G_i, G_j \in \mathcal{G},\ i < j\}$ is a set of directed arcs linking pairs of elements of $\mathcal{G}$. Each arc (G_i, G_j) is labeled by a face f_j of G_i, meaning that G_j is the refined map of the region represented by f_j in G_i; each such face will be called a *macroface*.

If (G_i, G_j) is an arc of $\mathcal{H}$, then G_i is called *parent* of G_j and, conversely, G_j is called *child* of G_j. The face f_j of G_i which defines the hierarchical relation is called *abstraction* of G_j and, conversely, G_j is called the *direct refinement* of f_j. Given a PEG G_i we call *descendants* of G_i in $\mathcal{H}$ all PEGs $G_j \in \mathcal{G}$ such that there exists a path in $\mathcal{H}$ from G_i to G_j.

Each PEG G_i in $\mathcal{G}$ is the representation of a map on a portion of the whole domain, at a given level of resolution $\varepsilon_h \in \tilde{\varepsilon}$: this means that all entities belonging to such a portion of the domain, that appears at resolution ε_h, are represented in G_i, and that the minimum precision among all edges (and faces) of G_i is ε_h. In particular:

- G_0 represents the map over the whole domain at resolution ε_0;
- if $(G_i, G_j) \in \mathcal{E}$ and G_i is at resolution ε_h, then G_j is at resolution ε_{h+1};
- all leaves of $\mathcal{H}$ are at resolution ε_k.

The relationship between a macroface and its refinement is defined by the following *hierarchy rule*, that is based on the concepts introduced in the previous Section:

> let $(G_i, G_j) \in \mathcal{E}$ be labelled by a macroface f_j of G_i at a resolution ε_h; the direct refinement G_j of f_j is a PEG whose domain is ε_h-homotopic to f_j, such that the region r approximated by f_j is ε_{h+1}-homotopic to the domain of G_j, and such that all new regions and features that appear in G_j have a minimum reolution level of (at least) ε_{h+1}.

If face f_j contains lineal features, then they will also be refined at precision ε_{h+1} into edges of G_j. We must impose that refinements of features and the refinement of the boundary of f_j do not cross. in order to maintain the topological consistency of G_j. The increase in detail will further subdivide the domain of G_j into faces, that will possibly contain new features not present in f_j.

In order to maintain the consistency of the whole hierarchical structure, we must enforce adjacent faces of a PEG to be refined into PEGs that "match" along their common boundary. This is guaranteed by the following *matching rule*:

> given a pair of macrofaces f_m, f_n of a PEG $G_i \in \mathcal{G}$, that share edges $e_1, \ldots, e_k$, the refinements G_m and G_n of f_m and f_n, respectively, will both contain the same refinements of $e_1, \ldots, e_k$, and will not intersect anywhere else: this property shall be maintained in the whole hierarchy, for all pairs of descendants of f_m and f_n that share a common portion of boundary.

In summary, given a map and a sequence of levels of resolution, our model explicitly encodes and relate different representations of such a map at all such resolutions. The relationships between different representations of an entity are established by the hierarchy rule, while the consistency of the representation of the whole map at each level of resolution is established by the matching rule.

5. A DATA STRUCTURE FOR ENCODING AN HPEG

An HPEG $\mathcal{H} = (\mathcal{G}, \mathcal{E})$ is encoded in a data structure combining individual data structures for representing the PEGs of $\mathcal{G}$, information describing hierarchical links defined by $\mathcal{E}$, and information needed to maintain the matching between PEGs sharing a portion of boundary.

We first describe the data structure for representing each component of the hierarchy, that was proposed in (De Floriani, Marzano and Puppo 1993). The data structure maintains three lists of entities: vertices, edges, and faces of the PEG, respectively. For each entity, some information about other entities topologically related to it are encoded:

- For each vertex:
  - its two coordinates;
  - if the vertex is isolated, a pointer to the face containing it, otherwise, a pointer to one of the edges incident into it;
- For each edge:
  - a pointer to its geometry (a chain of joints);
  - two pointers to its endpoints (vertices);
  - two pointers to its incident faces (in case the edge is a lineal feature, the two faces will be coincident);
  - four pointers to the first adjacent edges met by rotating counterclockwise and clockwise about its two endpoints, respectively.
- For each face:
  - a pointer to a list of edges, containing one edge for each connected component of its boundary;
  - a pointer to a list of edges, containing one edge for each connected component of its contained edges (lineal features);
  - a pointer to a list of isolated points contained in the face.

For each component G_j in an HPEG $\mathcal{H} = (\mathcal{G}, \mathcal{E})$, we must also encode information about the boundary of G_j, in order to support navigation in the hierarchy from G_j to any other PEG sharing an edge with it. Thus, for each $G_j \in \mathcal{G}$ we store the lists $l_1, \ldots, l_k$ of edges on the boundary of G_j, one list for each edge $e_1, \ldots, e_k$ of the face f_j refined by G_j, in counterclockwise order, respectively: in practice, each list l_i $(i = 1, \ldots k)$ is composed of pointers to the boundary edges of G_j refining a boundary edge e_i of f_j (see Figure 5).

Finally, we encode hierarchy information through parent-child links, i.e., for each macro-face f_j, a link to the direct refinement of f_j is maintained, and for each PEG G_i of the hierarchy, a link to the abstraction of G_i is maintained.

In (De Floriani, Marzano and Puppo 1993) a set of efficient accessing algorithms, that implement relational operators on the PEG has also been described. It is not difficult to extend such accessing algorithms to the HPEG, by exploiting additional information provided by the extended data structure.

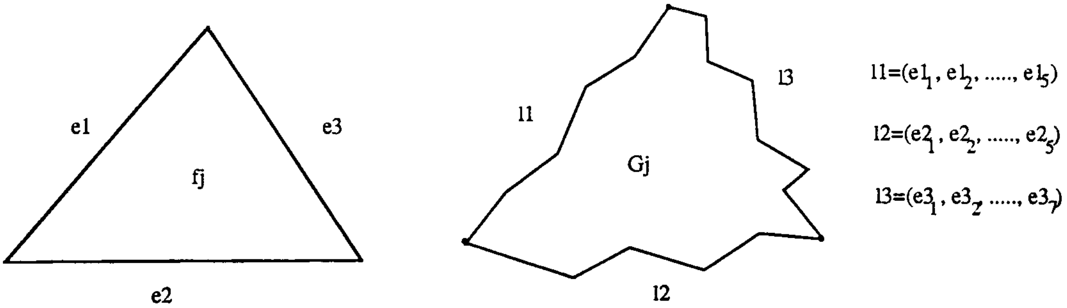


Figure 5: A region f_j is refined into a PEG G_j; for each edge e_i of f_j a corresponding list l_i of boundary edges of G_j refining e_i is maintained.

6. BUILDING AND MANIPULATING AN HPEG

The HPEG representing the multiresolution map of a given area can be built based on a sequence of levels of resolution $\bar{\varepsilon}$ defined in the previous Section, given a criterion for relating the relevance of entities in the map to each level of resolution.

The algorithm for building the structure can follow a top-down technique similar to the one described for the HTIN in (De Floriani and Puppo 1992): given a PEG G_i at resolution ε_h, its edges having a precision worse than ε_{h+1} are refined first, thus increasing the precision; then, faces induced by such edges are further refined independently by introducing new vertices and edges (that yield subfaces), thus increasing the detail.

An alternative algorithm following a bottom-up approach, builds the HPEG through recursive merging of maps at a lower level of resolution. In this approach, at each recursion level, entities and features can be discarded according to their relevance, thus reducing the detail; then, representations of remaining lines can be coarsened through a line-simplification algorithm.

In both cases, suitable computational techniques must be developed, that permits fulfilling the matching rule during construction.

From an HPEG, it is possible to extract expanded representations of the map of a given region, at a given detail, through an algorithm that is a generalization of the algorithm for expanding an HTIN, described in (De Floriani, Mirra and Puppo 1993).

One of the most important applications of a multiresolution map representation is in answering spatial queries and browsing over the map at different levels of resolution. In order to support such tasks, it is necessary to develop suitable algorithms for solving geometric problems like point location and line intersection, and for navigating the hierarchy.

Based on our former experience in solving spatial queries on the PEG (De Floriani, Marzano and Puppo 1993) and on the HTIN (De Floriani et al. 1994), we are currently developing query algorithms on the HPEG. Note that problems like point location become more complex on the HPEG because of the loose geometric correspondence between a face and its refinement: a point belonging to macroface f_j at precision ε_h could be external

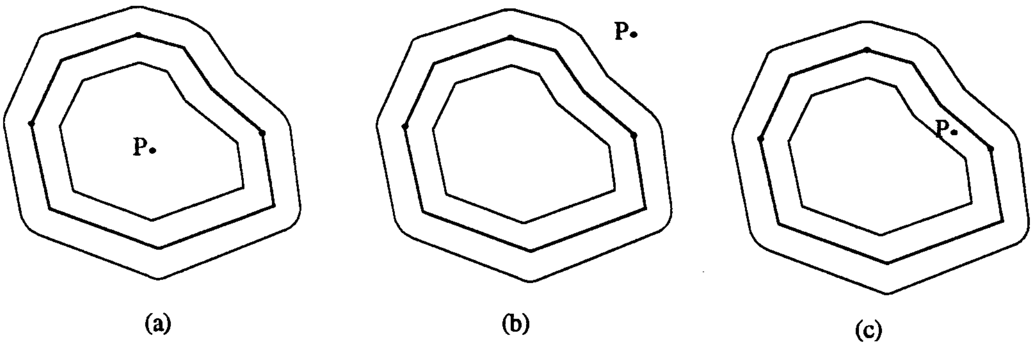


Figure 6: Hierarchical point location by using the kernel and the fattened region: (a) point p is in the kernel; the search continues in the refinement of f; (b) point p is outside the fattened region; the refinement of f needs not be searched; (c) point p is within the two regions; both the refinement of f and of its adjacent regions must be searched.

to the domain of its refinement G_j at precision ε_{h+1}. Thus, we solve point location by exploiting the ε_h-fattened region and the ε_h-kernel of f_j: if a point lies in the ε_h-kernel of f, then it certainly lies in the region represented by f_j; if it lies outside the ε_h-fattened region of f_j, it is certainly external to the region represented by f_j; otherwise, the search must be propagated both to the refinement of f_j, and to the refinements of regions adjacent of f_j and intersecting the ε_h-fattened region f_j. Such an approach permits to exploit the hierarchical structure of the HPEG to speed-up point location through a top-down search (see Figure 6).

7. CONCLUDING REMARKS

We have proposed a formal model that supports multiresolution representation of maps based on topological structures, together with a data structure for efficiently encoding such a model. Based on our former experience on multiresolution spatial models, we are currently studying and developing algorithms for construction, manipulation, and analysis of the model described here.

Throughout the paper, we have made some assumptions for the sake of simplicity, that are not always verified in practical cases. This fact leaves some problems open, that shall be considered in the future.

A first assumption is that in the convolution of a simple line s, the value of ε is always small enough to make $R_{s,\varepsilon}$ a simply connected region having s as skeleton. Similarly, we have assumed that for the contour c of a simply connected region, $R_{c,\varepsilon}$ is always a ring-shapd region, and that for a non-simply connected region with boundaries $c_o, \ldots, c_k$, regions $R_{c_0,\varepsilon}, \ldots, R_{c_k,\varepsilon}$ will never intersect. Such assumptions are somehow related with the fact that spatial entities should never be too dense in space or too close to one another, relative to the current precision: this is a fuzzy relation between precision and detail, that brings us back to the problem of map generalization. Anyway, since general cases might arise in real examples, in which ε is large enough to violate our assumptions (see

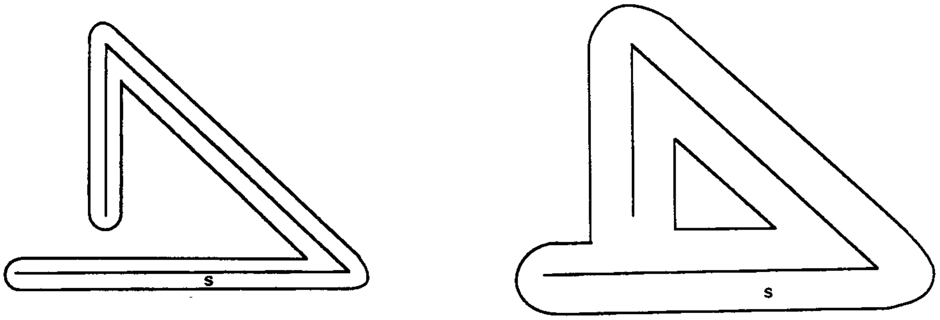


Figure 7: Two convolutions of the same line with discs of different radius: the convolution on the right is not a simply connected region.

Figure 7), it is worthwhile to consider methods to detect such situations and resolve them. Computational issues about such cases have been studied in (Guibas et al., 1993), together with other interesting problems related to the simplification of lines and subdivisions.

A second assumption is the fact that endpoints of open lines remain fixed under homotopy. Although this assumption coincides with the standard definition of homotopy of open lines from algebraic topology, in our application it is equivalent to assuming that vertices in the model are given without any error, i.e., their coordinates are exactly coincident with the coordinates of points they represent in the real world. In order to allow error on vertices, the definition of homotopy should be slightly modified by using a homotopy disc centered at each vertex: the endpoints of a line s might not be coincident with the endpoints of the chain e approximating it, but they must remain inside homotopy discs. On the other hand, this fact would make refinement more complicated: while in our assumption different chains incident in a vertex can be refined independently, if the vertex is allowed to move inside its homotopy disc suitable constraint must be introduced to ensure that all such chains are refined consistently.

The final and strongest assumption that we have made is that there exists a unique criterion to relate the relevance of each entity in a map with the precision of the representation, and that such a criterion can be applied automatically. As we pointed out earlier, such issue is not fully clear, as it is related to the unsolved problem of map generalization. Some heuristics could be adopted, that establish the relevance of an entity based on combined information about its size and the semantics associated with it. Otherwise, a semi-automatic approach can be followed, in which the entities to be discarded/inserted at each level of resolution should be selected manually or decided on the basis of some classification provided by the user, while the geometric simplification/refinement of lines and regions would be carried on automatically.

Although our hierarchical model is essentially region-oriented, issues in map generalization suggest that also other forms of refinement/abstraction should be faced. It is often useful to take into account the possibility that a given entity changes its dimension when the level of detail increases: for example, an entity that is represented by either a single point, or a line, at a given resolution is refined into a region at a higher resolution. Apart from further problems in the relation between precision and detail discussed above, this latter approach involves important changes in the model. Indeed, distinct hierarchical representations for entities of any dimensions (regions, lines, and even points) must be maintained, and such representations must be related consistently. Models that can support such a representation are subject of our current research, and will be developed in the future.

REFERENCES

De Floriani, L., Puppo E., "A hierarchical triangle-based model for terrain description" (1992) in *Theories and Methods of Spatio-Temporal Reasoning in Geographic Space*, eds. A.U. Frank, I. Campari, U. Formentini, *Lecture Notes in Computer Science* **639**, Springer-Verlag, pp.236-251.

De Floriani, L., Marzano, P. Puppo, E., "Spatial queries and data models" (1993) in *Spatial Information Theory - A theoretical basis for GIS*, eds. A.U. Frank, I. Campari, *Lecture Notes in Computer Science* **716**, Springer-Verlag, pp.113-138.

De Floriani, L., Mirra, D., Puppo, E., "Extracting contour lines from a hierarchical surface model" (1993) *Computer Graphics Forum* **12**, 3 (EUROGRAPHICS'93 Conference Issue), pp.249-260.

De Floriani, L., Gattorna, G., Marzano, P., Puppo, E., "Spatial queries on a hierarchical terrain model" (1994) in *Proceedings 6th International Symposium on Spatial data Handling*, Edinburgh (UK).

Egenhofer, M., Frank, A.U., Jackson, J.P., "A topological data model for spatial databases" (1989) in *Proceedings First Symposium, SSD'89 - Design and Implementation of Large Spatial Databases*, eds. G. Goos, J. Hartmanis, *Lecture Notes in Computer Science* **409**, pp.271-286.

Frank, A., Kuhn, W., "Cell graph: a provable correct method for the storage of geometry" (1986) in *Proceedings 2nd International Symposium on Spatial data Handling*, Seattle, WA.

Guibas, L.J., Hershberger, J.E., Mitchell, J.S.B., Snoeyink, J.S., "Approximating polygons and subdivisions with minimum-link paths" (1993) *International Journal of Computational Geometry and Applications* **3**, 4, pp.383-415.

Molenaar, M., "Single valued vector maps - a concept in GIS" (1989) *Geo-Informationssysteme* **2**, 1.

Peucker, T., "A theory of the carthographic line" (1976) *International Yearbook of Carthography* **16**, pp.134-143.

Pigot, S., "A topological model for a 3D spatial information system" (1992) in *Proceedings 5th International Symposium on Spatial Data Handling*, Charleston, SC.

Worboys, M.F., "A generic model for planar geographic objects" (1992) *International Journal of Geographical Information Systems* **6**, 5, pp.353-372.

Query Languages

IDENTIFICATION OF A DEFINITION FORMALISM FOR A SPATIAL VIEW

C. CLARAMUNT [1], M. MAINGUENAUD [2]

[1] Swiss Federal Institute of Technology,
Rural Engineering Department,
1015 Lausanne, Switzerland
E-mail: Christophe.Claramunt@dgr.epfl.ch
Telefax: 41-21-693-37-39

[2] France Télécom,
Institut National des Télécommunications,
9 rue Charles Fourier, 91011 Evry, France
E-mail: Michel.Mainguenaud@int-evry.fr
Telefax: 33-1-60-76-47-80

ABSTRACT. Today spatial database users need dynamic and flexible query and visualization languages. The objective of this paper is to propose an extension of the classic view concept as found in database models to meet these requirements. Our research relies on a spatial data model and on a visualization query language which could provide a basis for the identification of a spatial view formalism. We will propose and illustrate the spatial view definition as a dynamic tool and as a function of users' points of view.

1. INTRODUCTION

The proposed research attempts to resolve visualization needs by defining a spatial view concept for qualitative and personalized viewing of geographic information in function of its meaning. To reach this end, our approach proposes to extend the view concept, as it is defined for database models, to spatial information characteristics.

A view was first defined in the relational DBMS world as a virtual relation derived from one or several relation (e.g. Ceri 1991, Bertino 1992). An object-oriented view is similarly defined as a virtual class derived by an object-oriented query (Kim 1989) or as a virtual, possibly restructured, subschema graph of the global schema (Tanaka 1988). Object-oriented views allow the expression of external schemas derived from the logical level, while integrating relative dynamics within their definition (operators) (Bertino 1992). Generally, a view allows the representation of data according to a point of view and in function of objectives which are different from those of the schema. Its role may be seen as a flexible and evolutive representation; flexible in that it allows the user to choose the appropriate representation, and evolutive because it contributes to the correct cohabitation of the database schema and applications over time. Views are thus a favored means for dynamic expression of adaptable and evolutive forms of a given data set.

Unfortunately, no view definition has been offered for spatial data. This paper will propose an extension of the view concept to spatial data. We will define a spatial view as the necessary integration, for spatial data, of spatial and visualization operators in a classic view model. We will develop this approach by a preliminary identification of a general model for spatial data.

We begin this study with an overview of visualization query languages and by examining the state of the art to evaluate current proposals related to the spatial view concept. We will attempt to identify the characteristics of a spatial database, a reference framework among the spatial database models will be identified in order to facilitate further research. We will analyze the objectives and mechanisms which lead to the representation of a set of spatial information; identify the semantics of this heterogeneous information and their visualization operations which are the constituent elements of a spatial view. Interpretation factors for spatial views will be addressed, which will lead us to describe the diversity of information components to be represented and their interrelations, the elaboration of a graphical interface and the presentation constraints which interfere with spatial information visualization processes. Having identified these elements, we may then propose a language for spatial view definition.

2. RELATED WORK

The deficiencies of the SQL query language for spatial applications are today well known (Egenhofer 1992). One of the weaknesses of this language is the lack of visualization functions for spatial information (Frank 1982, Ingram 1987, Raper 1991, Egenhofer 1991). Graphic display of geographic features involves two issues: specifying the information to be displayed and the format in which to display it (Ingram 1987). Some attempts have been made to introduce requirements for graphical presentation in a spatial SQL (Frank 1982, Egenhofer 1991). Egenhofer proposes a Graphic Presentation Language which contains commands to direct, among others, the display of objects, spatial context, the query window, map scale and, taken as a whole, how to display query results.

The concept of a spatial view, or the idea associated with it, is indirectly addressed in graphical applications and hardly addressed in the area of spatial database research. Beard identifies the applicational, structural and graphical constraints which are associated with generalization processes and which underscore the need for a generalized view of a spatial database (Beard 1991). For graphical applications, the need for flexible visualizations to manipulate graphical data has been identified. Egenhofer points out that the choice of visualization must be controllable so that users may choose the visualization style which is appropriate for their application needs; this dynamic flexibility provides an essential advantage with respect to traditional maps (Egenhofer 1991). The term multiple view is proposed to designate a graphical layer for controling the visualization of arbitrary objects (Haarslev 1990); the view notion is also used to designate the semantic regrouping of multiple numerical maps (Tanaka 1988). Abel identifies the interest of a form of spatial view for geographic applications (Abel 1992). He notes that even if modeling tools such as the entity-relationship formalism, which allows the integration of different points of view, are useful, they cannot resolve all the problems of a large organization which represents a dynamic context concerning data representation, and notably conflicts between highly varied points of view. All these research efforts have indirectly addressed the spatial view concept. We will pursue similar objectives but with a database approach which will lead to a more formal definition.

3. A SPATIAL MODEL FRAMEWORK

3.1. Spatial data semantic

We will develop the spatial view proposal as an extension of the relational view, considering that the relational model is well known and provides a vehicle to present our ideas. A semantic approach to relations and attribute specifications may help the management of the results of spatial visualization operations. The quality of a spatial data model successively conditions the expressiveness of query and visualization languages. In particular, the characteristics of a spatial entity require the definition of relation and attribute consistency:

- *spatial overlap* of two instances of a same spatial entity (e.g. two overlapping flood zones) equivalent or not to a spatial partition (Mainguenaud 1993);

- *classic attribute domains* (e.g. integer, real, character) and *value modes* (ordinal, nominal, cardinal) defined by Stevens (1946). A spatial entity attribute has additional validity conditions. The *granule* defines the attribute validity for the *integrality* or for the *sub-set* of the spatial entity (e.g. an average altitude attribute is defined for the integrality and not for a subset of a spatial entity). The *topology* defines the topological validity (*interior*, *boundary*, *global*) of an attribute (e.g. a crop attribute is defined for the interior of a parcel).

These conditions may be specified as in the following example :

```
Create Table non_overlap City
(Name: string, (sub-set, global), primary key;
Population: string, (integral, interior))
Spatial-Representation: Spatial-Rep)
```

In this example, a city is a non overlapping spatial entity (one city at one geographic location). The city name is available for the globality (interior and boundary) and is still valid for a subset of the spatial entity. Conversely, the city population is available for the integrality (not for a subset) and for the interior of the spatial entity. We specify the spatial representation of the city by a special type (Spatial-Rep) which will be defined in the next sub-section. A similar framework may be used for query and visualization operators to propose a semantic control of the resulting alphanumeric data.

3.2. A complex object approach for spatial information description

We propose to define the spatial representation of an entity by a *"complex object"* definition (Adiba 1987). A complex object allows the extension of basic data types (e.g. integer, real, character). A complex object attribute may be decomposed into a set of component attributes. Attribute decomposition can be iterated as long as needed to ensure a complete description of an object (Parent 1989). This structure has some similarities with the encapsulation mechanism defined by the abstract data type. The latter is at the basis of the encapsulation mechanism defined within object-oriented databases (Stonebraker 1983). Complex objects allow the representation of static properties of spatial entities. We will use complex object notation, presented by the following notation, for the definition of the spatial representation of an entity:

> An atomic type is noted A: T where T is a type (e.g. integer, real, spatial domain). Types are recursively defined as follows:
>
> if T1, ..., Tn are distinct types and A an attribute then A: [A1: T1, ..., An:Tn] is a tuple of name A;
> if T1 is a type and A an attribute then A: {T1} is a set type of name A;
> if T1 is a type and A an attribute then A: <A1: T1, ..., An: Tn> is a list type of name A;
> if T1 is a type and A an attribute then A: (value(T1), ..., value(T1)) is an enumerate type of name A.

We characterize the elements of each spatial entity by a *spatial representation* (Spatial-Rep). Each spatial representation groups the cartographic primitives and their properties for each spatial entity :

- *cartographic primitive* (Cartographic-Prim): cartographic primitive defined on a spatial domain; it allows the spatial expression of a spatial entity. Several authors propose a spatial domain definition with different semantic complexities (e.g. Egenhofer 1991, Worboys 1993). We will consider, for our demonstration, a spatial domain specified by a complex object called geom without considering its structure;

- *observation scale* (Observation-Scale): property which specifies the spatial referencing scheme or the perception level of the phenomenon (e.g. global, regional, local) within which the cartographic primitive is defined (adapted from Golay 1992);

- *validity scale* (Validity-Scale): property which defines the scale validity of the cartographic primitive;

To complete these definitions, the *significant scale* (Significant-Scale) specifies and references the observation scale for which the spatial entity attributes are semantically interpreted.

The complex object of the spatial representation is then defined by:

> Spatial-Rep:
> [
> <
> [Cartographic-Prim: geom;
> Observation-Scale: string;
> Validity-Scale:[Limit-Min: real; Scale-Capture: real; Limit-Max: real]

```
]
>
Significant-Scale: integer
]
```

If the complex object of a spatial representation has at least two cartographic primitives, then the corresponding spatial entity has a multi-valued spatial representation or, in other words, a multiple spatial representation. A spatial entity may have many cartographic primitives for a considered scale if their validity scale intervals overlap. In this case, the spatial entity has multiple interpretations for a same scale (e.g. two different user views for the same spatial entity).

For each spatial entity, the spatial representation defines and manages the cartographic primitives and their properties. This specification of the properties at the spatial entity level allows data representation in a heterogeneous application situation (i.e. many users and semantics for same spatial entities).

3.3. Real or virtual representation

A spatial entity visualization may be realized from two types of cartographic primitive (adapted from Sacks-Davis 1987):

- *"real"* cartographic primitives which correspond to a geometric space reality and are significant for spatial entity attributes (e.g. a city polygon defining an urban area limit which is geometrically accurate);

- *"virtual"* cartographic primitives related to a spatial entity but not geometrically significant (e.g. a city polygon represented by a schematic limit which is not geometrically accurate).

A spatial operation is generally efficient if it is performed from real cartographic primitives of spatial entities. For instance, a spatial intersection operation has to be computed from a real cartographic primitive and not from a virtual representation, a function of a real cartographic primitive. Conversely, virtual representations do not guarantee the quality of the result of a spatial operation. However, a virtual representation may be used for approximate spatial operations in the case of complex or large databases, where it provides a preliminary evaluation result. Operator validity rules are inferred from real or virtual types of cartographic primitives and thus independently of spatial entity semantics. Therefore, the complex object definition of a spatial entity may be extended by a the definition of a mode, which specifies the real or the virtual type of a cartographic primitive:

```
Spatial-Rep:
[
<
[Cartographic-Prim: geom;
Observation-Scale: string;
Validity-Scale: [Limit-Min: real; Scale-Capture: real; Limit-Max: real]
```

Mode: (real, virtual)
]
>
Significant-Scale: integer
]

3.4. Visualization variables and functions

We have developed a static representation model for spatial information. This model allows the identification of the entities implicated in an SQL query (from clause) and a relative evaluation of operation applications (where clause). We will now extend this reference model by a description of the visualization parameters and functions which will lead to a management of the graphic presentation of query results (within the select clause).

Visualization processes are a function of spatial representation and particularly of the different cartographic primitive levels. The spatial representation definition permits the encapsulation of identified and stable visualization parameters. As we consider the spatial view as a query interface and not as a cartographic tool, this definition level is sufficient. Therefore, the system is told how the values (spatial entities in this case) are to be displayed (Osborn 1986). These parameter visualizations are classified and referred to as Bertin variables (Bertin 1983). These variables allow the integration of visualization parameters in a spatial query language (Egenhofer 1991). Combination of these variables provides a visualization operation with sequential parameters such as color, value (gray degree or intensity), pattern (graphic symbol), size (e.g. of a graphic symbol), texture (e.g. pattern resolution) and orientation (e.g. of a graphic symbol). These Bertin variables are applied to cartographic primitives (e.g. a polygon may have an exterior visualization color and an interior symbol pattern). For our study, we will define the Bertin variables as a complex visualization object (Bertin-Variable) which is integrated in the spatial representation:

Spatial-Rep:
[
<
[Cartographic-Prim: geom;
Observation-Scale: string;
Validity-Scale: [Limit-Inf: real; Scale-Capture: real; Limit-Sup: real]
Mode: (real, virtual);
Visualization: Bertin-Variable
]
>
Significant-Scale: integer
]

In order to extend visualization capabilities, visualization functions complete the graphic presentation language (Ingram 1987). Many authors propose visualization functions for spatial information which may be logically inserted in the select clause of a query:

- generalization operators (Brassel 1987, Beard 1991) such as simplification, smoothing, aggregation, amalgamation, merging, collapse, refinement, exaggeration, enhancement and displacement (McMaster 1991);

- thematic classification operators in accordance with cartographic methods (Huang 1993);

- display and position operators for textual information managed by specialized algorithms (Freeman 1987);

- context operators such as legend display (Egenhofer 1991).

This overview of visualization functions aims to illustrate our purpose. This shows the interest of visualization functions within a spatial query language. However, visualization functions for query languages have to be limited to single operations so as to avoid complex manipulations which may change the nature of query objectives. A visualization process creates new entities defined as the projection of spatial entities by query and visualization operations or, in other words, as the visualization domain of the visualization operation. This visualization process analysis may be continued by an evaluation of the management of constraint presentations defined as the association of windowing parameters and human visualization capacity (Robinson 1984). This allows the user to interact with the graphic interface and improve the quality level of the application (Voisard 1991). The specification within this model of the spatial and visualization properties gives the user an aid-oriented approach which facilitates visualization operations compared to a free solution where the user has to choose all visualization parameters.

4. A SPATIAL VIEW CONCEPT PROPOSAL

We have defined a spatial representation model and analyzed parameter and function visualizations. We propose an approach to spatial view definition from these static and dynamic elements. A spatial view is designed as the application of spatial and non spatial queries (O) combined with spatial visualizations (OV) of one or many spatial and non spatial relations. Together, the query and visualization operations form a manipulation language. A collection (C) is defined as a set of spatial and non spatial instances used within a spatial view creation. This definition is related to the horizontal view proposed by Hegner (Hegner 1991) in that the spatial view is designed as a dynamic grouping of different complementary entities.

Each triplet [{C}, {O}, {VO}] defines a spatial view atom. A spatial view atom is a member of at least one spatial view. A spatial view atom is a visualized spatial relation and is derived from a collection set. A spatial view atom may be defined as the application of spatial and non spatial operations combined with visualization operations on a collection set. Spatial view atoms are closed under their definition. A basic spatial relation is defined as a spatial view atom with empty spatial and non spatial operations and an empty set of visualization operations. A spatial view groups together spatial view atoms. It is defined as a relation which is both spatial and non spatial and may be composed of relations and derived relations. Unlike the relational view definition, a spatial view is not a relation but a set of relations.

Each spatial or non spatial entity projected in the realization process of the spatial view will be a member of a spatial view atom decomposed into (1) a collection set of spatial and non spatial entities (2) a query and visualization set and returning (3) a visualization entity set. Figure 1 shows a comparison of view and spatial view realization processes:

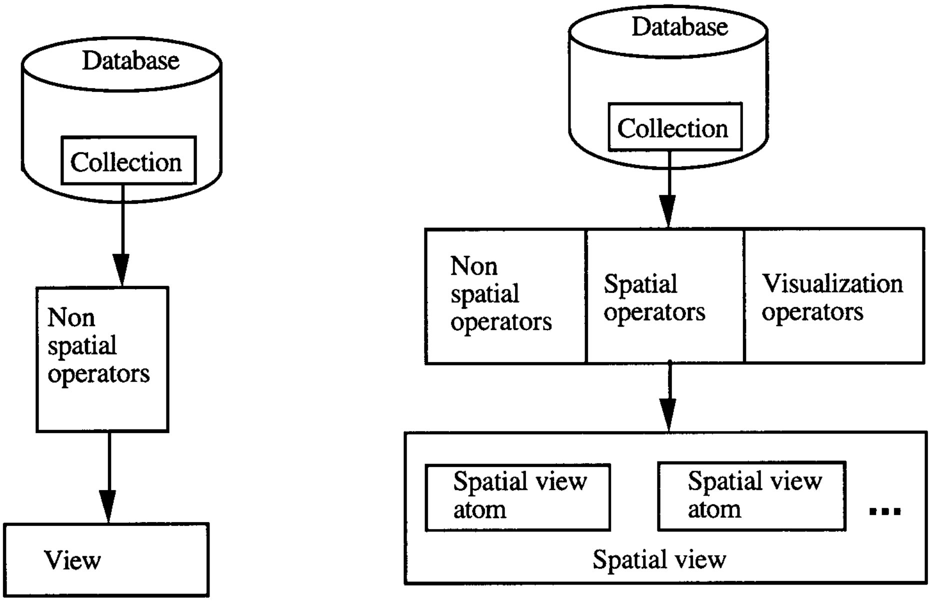


Figure 1: View and spatial view realization

A same spatial or non spatial entity may be member of different collections, different spatial view atoms and many spatial view realization processes. To each virtual or real visualization entity, correspond one-to-many entity references from a collection used by the spatial view process.

Human perception of graphic information is limited by visual acuteness and size limitations of a graphic interface, especially in the case of complex queries. Moreover, when the scale is reduced, linear entities are overestimated with respect to polygonal entities, or visualized classes may be reduced and the visual density may increase. These progressive and interrelated phenomena have to be controlled to ensure the quality of visualization results. The different visualization collections are semantically and cartographically interdependent. Visualization entities may be ordered by relative semantic importance and relative query roles. This can be seen as an answer to both application objectives and the progressive saturation of a display space query. The respective contribution of each spatial view atom may be consequently analyzed depending on the query semantics with respect to user and application objectives. We can distinguish three level of collection importance for spatial view atoms:

- collections directly involved in the query result (essential);

- collections used within the query (important);

- collections used as background maps (useful) (Ingram 1987).

We will represent, within the spatial view definition, this classification by an order attribute which manages the relative importance of the different spatial view atoms (essential, important, useful). Each spatial view will be additionally defined by a visualization scale (bounded and referenced) which will be specified by a visualization scale attribute. Therefore a spatial view is defined as:

```
Spatial-View:
[
Name;
Visualization-Scale: {[Limit-min, Scale-Display, Limit-Max]};
Order: [essential: {Spatial-View-Atom};
        important: {Spatial-View-Atom};
        useful: {Spatial-View-Atom}]
]
```

To explain this concept, we propose a basic example of a spatial view design. In particular we will try to show the relative importance of each spatial view atom. A first classification of spatial view atoms displays the spatial representation of Portuguese cities which have more than 100 000 inhabitants (essential order):

C1, a spatial entity collection:
C1: {City, Country}

```
Create Spatial-View-Atom Classification
select  City.Spatial-Rep
        display text City.Name
from City, Country
where City.Population > 100 000
and Country.Name = 'Portugal'
and City.Spatial-Rep inside Country.Spatial-Rep
```

Two additional spatial view atoms, Country (important order) and River (useful order), complete the first spatial view atom and provide a complement for the spatial view:

C2 a spatial entity collection:
C2: {Country}
C3, a spatial entity collection:
C3: {River}

```
Create Spatial-View-Atom Country
select Country.Spatial-Rep
from Country
where Country.Name = 'Portugal'
```

```
Create Spatial-View-Atom River
select River.Spatial-Rep
from River
```

```
where River.Length > 300
and Country.Name = 'Portugal'
and River.Spatial-Rep intersect Country.Spatial-Rep
```

The explicit designation of the spatial view atoms identifies the different related queries of the spatial view. The spatial view example is finally defined as :

```
Create Spatial View
[
Name: example;
Visualization-Scale: [1/20 000, 1/10 000, 1/5 000]
Order: essential: [Classification]
       important: [Country]
       useful: [River]
]
```

This spatial view example shows a visualization composition which groups together the related spatial view atoms. The query meaning is explained by the Classification spatial view atom (essential order). The query result is dependent on the Country spatial view atom (important order) and the final visualization result is graphically completed by the River spatial view atom (useful order). This spatial view gives, among others, an idea of Bertin variable utility in the query visualization process for the different spatial entities visualized (pattern, size, symbol).

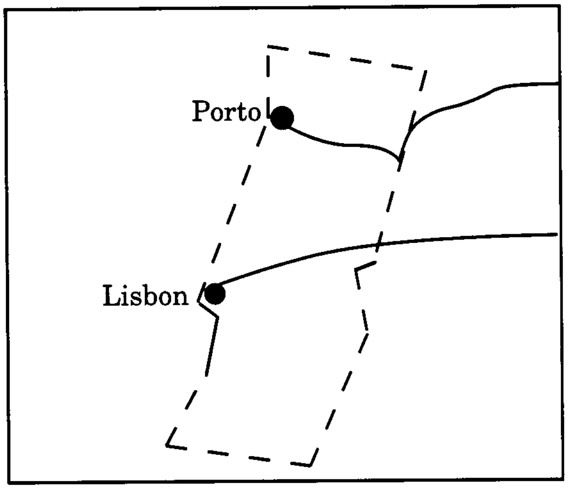


Figure 2: spatial view example

The semantic order defined within the complex attribute definition may be completed, at the interface level, by a logical order which controls the display priorities (e.g. polygons are generally displayed before lines).

5. CONCLUSION

The spatial view concept gives users flexibility to compose a graphic database view adapted to spatial application requirements. The spatial view may be suited for spatial database queries or even updates. From an external point of view, the spatial view concept allows different, independent external interpretations of database schemas. This provides a support for schema evolutions and a partial solution for application portability with respect to user needs.

We have proposed and justified a spatial view concept as an extension of the view developed within classic database models. We have identified static and dynamic constituents of spatial entities by a complex object specification. The spatial view is realized as a sequence of selection and visualization operations. This process is structured by spatial view atoms which are the incremental, basic elements of a spatial view definition. Spatial view atoms provide an ordered relation to characterize the relative importance of the visualized data.

At present we have several directions for our future research. The spatial view concept has to be defined exhaustively and formally. In particular, spatial view properties and manipulation capabilities are interesting development issues. Materialized or derived views provide two different implementation solutions which have to be analyzed with respect to application conditions for the spatial view case. For instance, a materialized spatial view has to be maintained through database evolution which leads to the study of recomposition rules and mechanisms. Integration of the spatial view concept towards a semantic network for hypertext navigation may give a user-oriented application interface.

Many research activities are related to an improvement of the spatial view. New extensions for the classic view concept, current developments in the area of visual query languages using metaphors or near natural languages are beneficial factors. Additional developments in constraint presentation issues are also a related development to this work.

ACKNOWLEDGEMENTS

We would like to thank Christine Parent and Marius Thériault for their helpful comments on the subject of this paper. We are also grateful to Laura Vidale for her attentive reading of preliminary versions.

REFERENCES

Abel, D. J., Yap, S. K., Ackland, R. G., Cameron, M. A., Smith, D. F. and Walker, F. G. (1992), Environmental Decision Support System Project: An Exploration of Alternative Architectures for Geographic Information Systems, in *International Journal of GIS,* vol. 6, n° 3, pp. 193-204

Adiba, M. E. (1987), Modeling Complex Objects for Multimedia Databases, in *Entity-Relationship Approach : Ten Years of Experience in Information Modeling*, eds. Spaccapietra, S., pp. 89-117

Beard, K. (1991), Constraints on Rule Formation, in *Map Generalization*, eds. Buttenfield, B. P. and McMaster, R. B., pp. 121-135

Bertin, J. (1983), *Semiology of Graphics*, University of Wisconsin Press

Bertino, E. (1992), A View Mechanism for Object-Oriented Databases, in *Proc.* of *Advances in Database Technology EDBT '92*, Vienna, March 23-27, pp. 136-151

Brassel, K. E. and Weibel, R. (1987), A Review and Conceptual Framework of Automated Map Generalization, in *International Journal of GIS*, vol. 2, n° 3, pp. 229-244

Ceri, S. and Widom, J., (1991), Deriving Production Rules for Incremental View Maintenance, in *Proc.* of *17th Int. Conf. on Very Large Data Bases*, Barcelona, September 3-6, pp. 577-589

Egenhofer, M. J. (1991), Extending SQL for Graphic Display, in *Cartography and GIS*, vol. 18, no. 4, pp. 230-245

Egenhofer, M. J. (1992), Why not SQL, in *Int. J. of GIS*, vol. 6, no. 2, pp. 71-85

Frank, A. U. (1982), MAPQUERY: Data Base Query Language for Retrieval of Geometric data and their Graphical Presentation, in *Computer Graphics*, vol. 16, n° 3, pp. 199-207

Freeman, H. and Ahn, J. (1987), On the Problem of Placing Names in a Geographical Map, in *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 1, n° 1, pp. 121-140

Golay, F. (1992), *Modélisation des Systèmes d'Information à Référence Spatiale et de leurs Domaines d'Utilisation Spécialisés; Aspects Méthodologiques, Organisationnels et Technologiques*, PhD Dissertation, EPFL, Lausanne

Haarslev, V. and Möller, R. (1990), A Declarative Formalism for Specifying Graphical Layout, in *Proc.* of *IEEE Workshop on Visual Languages*, Skokie, Illinois, October 4-6, pp. 54-59

Hegner, S. J. (1991), Pairwise-Definable Subdirect Decompositions of General Database Schemata, in *Proc.* of *Symp. on Mathematical Fundamentals of Database Systems*, Rostock

Huang, Z. (1993), Designing a Spatial Query Language for Spatial Data Query and Analysis, in *GIS Technology and Applications*, eds. Lu, H. and Ooi, B. C., Singapore, June 21-22, pp. 209-223

Ingram, K. J. and Phillips, W. W. (1987), Geographic Information Processing Using a SQL-Based Query Language, in *Proc.* of *Auto-Carto Conf. 8*, Baltimore, March-April, pp. 326-335

Kim, W. (1989), A model of Queries in Object-Oriented Databases, in *Proc.* of *International Conf. on Very Large Data Bases*, Amsterdam, August, pp. 423-432

McMaster, R. and Mark, D. M. (1991), The Design of a Graphical Interface for Knowledge Acquisition in Cartographic Generalization, in *Proc.* of *GIS/LIS'91*, Atlanta, October 28 - November 1, pp. 311-320

Mainguenaud, M. (1993), The Results of Geographical Information System Queries, in *Proc.* of *IEEE/CS Visual Languages '93*, Bergen, Norway, August 25-27, pp. 362-364

Osborn, S.L. and Heaven, T. E. (1986), The Design of a Relational Database System with Abstract Data Types for Domains, in *Proc.* of *A C M Transactions on Database Systems*, vol. 11, n° 3, pp. 357-373

Parent, C. and Spaccapietra, S. (1989), Complex Objects Modeling: An Entity-Relationship Approach, in *Nested Relations and Complex Objects in Databases*, Springer-Verlag, n° 361, pp. 272-296

Raper, J. F. and Bundock, M. S. (1991), UGIX: A Layer Based Model for a GIS Interface, in *Cognitive and Linguistic Aspects of Geographic Space*, eds. Mark, D. M. and Frank, A. U., pp. 449-475

Robinson, A. H., Sale, R. D. and Morrison, J. L. (1984), *Elements of Cartography*, ed. John Wiley, New York

Sacks-Davis, R., McDonell, K. J. and Ooi, B. C. (1987), *GEOQL - A Query Language for Geographic Information Systems*, Internal Report no. 87/2, Dept of Computer Science, Royal Melbourne Institute of Technology, Australia

Stevens, S., (1946), On the Theory of Scales of Measurement, in *Sciences Magazine*, 103, pp. 677-680

Stonebraker, M., Rubenstein, B. and Guttman, A. (1983), Application of Abstract Data Types and Abstract Indices to CAD Data Bases, in *Proc.* of *Engineering Design Applications of ACM IEEE Database Week*, Los Angeles, pp. 107-113

Tanaka, M and Ichikawa, T. (1988), A Visual User Interface for Map Information Retrieval Based on Semantic Significance, in *IEEE Transaction on Software Engineering*, vol. 14, n° 6, pp. 666-670

Voisard, A. (1991), Towards a Toolbox for Geographical User Interfaces, in *Advances in Spatial Databases*, eds. Günther, O. and Schek, H.-J., Zurich, August 28-30, pp. 75-98

Worboys, M. F. and Bofakos, P. (1993), A Canonical Model of Areal Spatial Objects, in *Advances in Spatial Databases*, eds. Abel, D. J. and Ooi, B. C., Singapore, June 23-25, pp. 36-52

APPROACHES TO THE REPRESENTATION OF QUALITATIVE SPATIAL RELATIONSHIPS FOR GEOGRAPHIC DATABASES

ALIA I. ABDELMOTY AND M. HOWARD WILLIAMS
Department of Computing and Electrical Engineering,
Heriot-Watt University
Riccarton, Edinburgh EH14 4AS, Scotland, UK
e-mail: <alia,howard>@cee.hw.ac.uk

ABSTRACT. Qualitative spatial reasoning is important for many spatial information systems, including GISs. It is based on the manipulation of qualitative spatial relationships and is used to infer spatial relationships which are not stored explicitly in the Geographic DataBase (GDB), to answer spatial queries given partial spatial knowledge and to maintain the consistency of the GDB. This paper compares the basic approaches developed for expressing qualitative topological, orientation and direction relationships. An extension of one of the classified approaches is then used for the representation of directional relationships (north, south, east, etc.) between objects of arbitrary shapes and for the representation of flow direction relationships (topological relationships between objects carrying flow, e.g. road segments, utility networks, etc.) as required within a GIS.

1. INTRODUCTION

Qualitative reasoning is an important tool for spatial databases for two reasons. Firstly, the common-sense reasoning which humans apply to spatial problems is generally qualitative; second, there is a growing need for the automatic derivation of qualitative spatial relationships to complement computationally expensive computational geometry.

Qualitative reasoning is based on the manipulation of qualitative spatial relationships. Representation formalisms for modelling qualitative spatial relations are being studied for topological, direction and orientation relationships. A number of formalisms have been put forward for representing qualitative relationships, and extensions to query languages have been developed to incorporate some of these.

In this paper, these approaches are compared from the point of view of the spatial interaction process which they adopt as well as the model of representation which they use. Comparisons of such approaches are useful in determining their advantages and current limitations, possible extensions and applicability to GISs.

One of the approaches reviewed has proved to be suitable for implementation in a GDB context, and as a result is currently adopted in the GIS community. In this paper, we extend this approach to include the representation of two other types of spatial relationships, namely, cardinal direction relationships, and flow direction relationships. The representation of both categories is important in the GDB and using the former approach to their representation makes the formalisms developed readily implementable in a GDB.

2. APPROACHES TO REPRESENTING QUALITATIVE SPATIAL RELATIONSHIPS

Any formalism for the representation of binary qualitative spatial relationships can be judged by its ability to satisfy the following criteria.

1. Completeness: where all the possible qualitative relationships between the objects under consideration are covered in the formalism, i.e., the formalism is exhaustive.
2. Soundness: where only a feasible or physically correct set of relations can be derived from the formalism.
3. Uniqueness of representation: where every qualitative relationship between objects considered can be uniquely distinguished in the formalism, i.e., the possible relationships are mutually exclusive.
4. Generality: where the formalism can handle objects of different shapes as well as different types of relations.

One way of classifying approaches for realizing such a formalism is by distinguishing between *Intersection-based* and *Interaction-based* approaches.

2.1 Intersection-Based Approach

In this approach an object is represented in terms of the set of its components, and relationships are the result of the combinatorial intersection of those components.

In Güting [23], objects are considered as point-sets and the usual set operators are used to define some topological relationships such as overlap and inside. However, other relationships such as neighbourhood cannot be defined using set operators and hence this method is incomplete.

In Egenhofer [8,12], point-set topology is used for the definition of the object components as interior (A°), boundary (δA) and exterior (A^-). Spatial relationships between the objects considered are the result of the exhaustive combinatorial intersection of their components represented in a matrix which is used as a model for representation. The approach does not assume any knowledge of possible relationships between objects, but derives all the possible (and impossible[1]) relationships through combinatorial exhaustion. It has been utilised mainly for the representation of topological spatial relationships where any qualitative topological relationship can be represented in this formalism. This approach is complete, although not sound. Also, there is no guarantee of uniqueness of representation if more than one qualitative relationship falls under the same matrix state [18], for example, in figure 1 two qualitatively different relationships are represented by the same intersections of their components (in both cases, an intersection exists between all the components of interiors, boundaries and exteriors).

Several extensions to the latter method have been developed with the aim of enriching the formalism, and hence moving towards uniqueness of representation. For example, Clementini [5] showed that by recording the dimension of the resulting intersections as well as considering the boundaries of a line object as a set of disconnected points, the approach could be extended to distinguish more qualitative relationships. Also in [18], more elaborate extensions have been adopted to include not only the dimension of the resulting components but also the number of occurrences of particular intersections.

The approach has been applied using regions and has been extended for the representation of relationships between lines [11] and regions with holes [15]. Egenhofer's approach and the various extensions thereof have been used to represent topological relationships only. For this purpose the approach has proved to be complete but not sound. In order to achieve soundness additional 'filtering rules' have to be added which are specific to the shapes involved. Nevertheless this approach has become popular in the GIS community and has been used in the design of spatial query languages and

---

[1] Egenhofer [9] discusses elimination of the impossible cases.

derivatives thereof have been implemented in commercial GISs [27].

2.2 Interaction-Based Approach

In this approach, the body is considered as a whole and is not decomposed into its components, as in the *intersection-based* approach. The set of possible relationships between the objects involved is pre-supposed (intuitively) and no guarantee of completeness is possible. However, a unique representation is developed for each possible relationship used. Since the shape of each object involved is usually pre-defined, the approach is strongly dependent on the shapes of the objects and the formalism has to be adapted whenever different shapes are considered.

The main approach in this category is due to Randell et al [31,33] and is based upon Clarke's calculus of individuals [4] which in turn is based on 'connection' where a set of topological relationships between concave regions are axiomatized. Attempts have been made to extend the formalism to approach *generality* by introducing different taxonomies of relationships between convex regions [6]. The formalism has also been extended to include the representation of orientation relationships in [7].

Other interaction-based approaches have been developed utilizing Allen's temporal logic by Freksa [20], Mukerjee & Joe [28] and Chang & Liu [3]. This is extended in Freksa [21] to include the representation of the qualitative orientation relationships where the relative orientation in 2-D is given by two oriented lines joining sets of two point objects. The approach is limited to the point-based abstraction of spatial shapes and different divisions of the space have to be developed if different shapes are considered. The latter approach has been extended in [34] to utilize qualitative distance (relative distance between objects). Hernandez [25,26] used a triangular division of space to represent a combined model for topological and orientation relationships. His model is also capable of handling different levels of granularity as well as transformation between frames of references (rotating labels). Orientation relationships represented in the above approaches are relative and can be used, for example, in way-finding (determining the position of object c with respect to ab by determining the orientation of the vector bc to vector ab).

In representing cardinal direction relationships, Frank [17] used a method of rectangular division of the embedding space (using projection) for the definition of direction areas which proved to produce more definite results in the propagation of cardinal relationships.

The above approaches can all be classified as 'Interaction-based' since objects are not decomposed into components, intersection operations are not used and the set of possible relationships in the domain are pre-supposed.

Note that if the domain is restricted to a set of simple (regular) shapes then both the *Intersection* and the *Interaction* approaches can be made complete and unique with respect to representation.

3. EXTENDING THE INTERSECTION-BASED APPROACH FOR THE DEFINITION OF DIRECTIONAL RELATIONSHIPS

Qualitative directional relationships are important spatial properties between objects in the geographic space that can be used as a powerful search constraint. Unlike other spatial relationships, such as adjacency or containment, direction is a fuzzy concept and is thus often dependent on human interpretation. The problem is made more complex with arbitrary shapes and sizes of objects as present in a GDB. In Peuquet & Xiang [29], the semantics of the directional relationships for arbitrary shaped objects is given, where the triangular model of representation examined by Haar [24] has been extended to take into account the size, shape and orientation of polygons.

In the triangular model direction can be determined by testing if the centroid of P_2 falls within a triangular area extended outwards from the centroid of P_1, as shown in figure 2(a). This model was shown to fail in different cases where, for example, objects are in close proximity in relation to their overall size, as shown in figure 2(b).

This model is extended in [29] to use the frame of the body in determining its relative direction. As shown in figure 3(a), the frame side of P_2 facing the other object P_1 is used to determine the vertex (called the centroid) of the triangulated area of acceptance. The vertex is moved backward or forward relative to the direction at issue, as shown in figure 3(b). In the case of intertwined objects, lines projected from the "centroid" of the objects are used to determine the relative direction as shown in figure 3(c).

In this paper, the above approach is used in defining the directional areas. No consideration is given to objects which are intertwined, i.e whose bounding boxes overlap. The above representation provides significant advantages over point-based abstraction of objects as adopted for example, in Freksa [21].

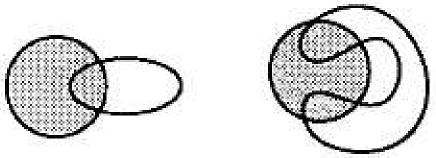


Figure 1: Non-unique representation where more than one relationship could represented by the same matrix state.

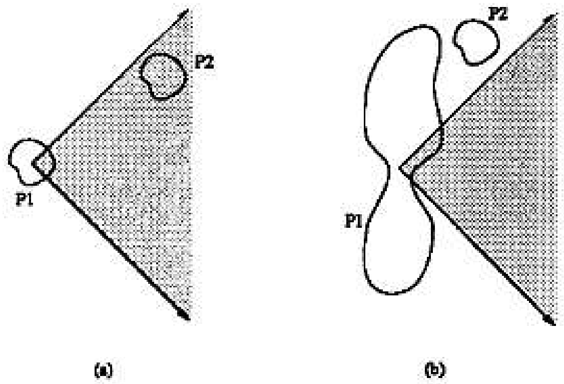


Figure 2: Using the triangular model for determining the relative direction between two objects. In (b) the model gives erroneous results for polygons in close proximity.

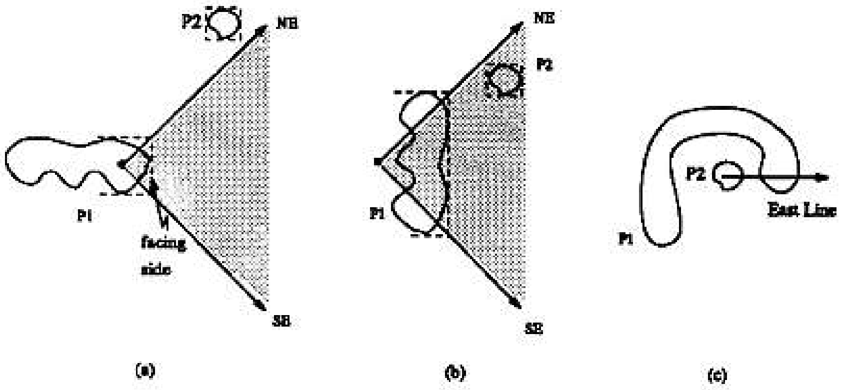
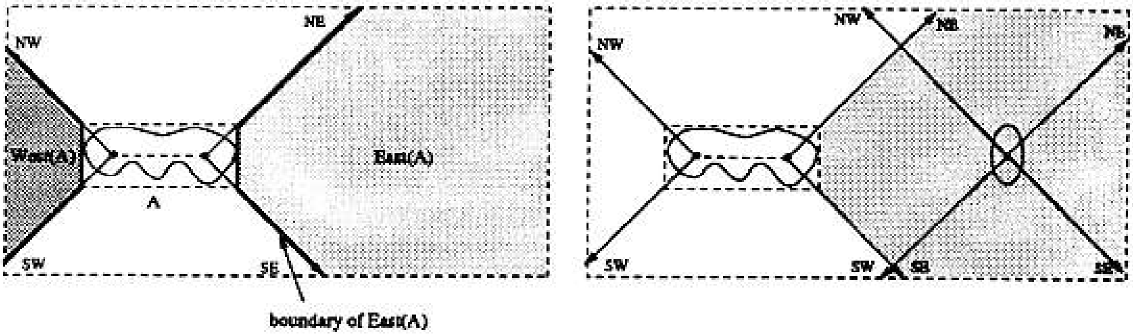


Figure 3: The modified triangular model by Peuquet & Xiang which takes into account the shape and orientation of objects. In (c) a line projected from the "centroid" is used to determine relative direction in convex and intertwined objects.



(a)

(b)

Figure 4: The four semi-infinite areas associated with objects in the space. The sides of the bounding box of the object form part of the boundaries of the semi-infinite areas in (a) and the intersection of the boundaries of these areas determine the relative direction in (b).

In the formalism developed here, every object is seen to divide the space into four semi-infinite areas as shown in figure 4(a). The exterior of an object is then considered to be the union of its four related areas.

The formalism follows the *Intersection-based* approach to representing the spatial relationships. In particular,

- Each of the four semi-infinite areas representing the object is represented through its components, namely, interior, boundary and exterior.

- The relations between any two objects are defined through the intersection of the components of the related areas.

The boundary of each area is represented by two directional lines (for example, NE *and* SE for the area representing east) and a characteristic point or one side of the bounding box of the object considered, as illustrated in figure 4.

In the simple case the bounding box of the body can be approximated to a line, (horizontal or vertical). The general intersection matrix is shown in figure 5. Here NE_1 is used to denote the North-East line for the first object, NE_2 the same line for the second object, B_1 the characteristic line of first object, and so on. Note that in the case of point abstraction of objects

$$\begin{bmatrix} NE_1 \bigcap NE_2 & NE_1 \bigcap NW_2 & NE_1 \bigcap SE_2 & NE_1 \bigcap SW_2 & NE_1 \bigcap B_2 \\ NW_1 \bigcap NE_2 & NW_1 \bigcap NW_2 & NW_1 \bigcap SE_2 & NW_1 \bigcap SW_2 & NW_1 \bigcap B_2 \\ SE_1 \bigcap NE_2 & SE_1 \bigcap NW_2 & SE_1 \bigcap SE_2 & SE_1 \bigcap SW_2 & SE_1 \bigcap B_2 \\ SW_1 \bigcap NE_2 & SW_1 \bigcap NW_2 & SW_1 \bigcap SE_2 & SW_1 \bigcap SW_2 & SW_1 \bigcap B_2 \\ B_1 \bigcap NE_2 & B_1 \bigcap NW_2 & B_1 \bigcap SE_2 & B_1 \bigcap SW_2 & B_1 \bigcap B_2 \end{bmatrix}$$

Figure 5: Intersection matrix for directional relationships where the body is vertical or horizontal represented by a characteristic line B_i.

the matrix is reduced to 4 × 4 where no characteristic line of the objects is considered.

Constraints such as parallelism of cardinal lines on each body restricts the combinatorial possibilities in the matrix. For example, the relationship $east(A, B)$ for the given case can be derived if the following intersections hold: ($NW_A \bigcap NE_B$ and $SW_A \bigcap B_B$) OR ($NW_A \bigcap B_B$ and $SW_A \bigcap B_B$) OR ($NW_A \bigcap B_B$ and $SW_A \bigcap SE_B$) OR ($NW_A \bigcap NE_B$ and $SW_A \bigcap SE_B$)

Let E_A, W_A, N_A, S_A denote the four semi-infinite areas (East, North, West, South) relative to body A. Let d_i denote the relative direction relation between two bodies, thus A $[d_i]$ B mean that body A is in the direction d_i with respect to B (where d_i is one of the cardinal directions, d_e for east, d_w for west, etc.).

Also let $(d_i)_A$ denote one of the four sides of the bounding box of body A as shown in figure 6, $\bar{d}_i$ denote its inverse, e.g. if $d_e = East$ then $\bar{d}_e = West$. Let $(d_{ij})_A$ represent one of the directional lines of the semi-infinite areas. For example, $(d_{ne})_A$ is $North\ East(A)$, and so on.

For a body B to be *east* of another body A, it must lie either totally or partially inside E_A. In this case W_B must overlap E_A resulting in a finite area[2]. The intersection of W_B and E_A can be represented by the intersection of their components, in particular, the boundaries of the two areas must intersect as shown in figure 4(B). Since the boundaries of the areas are each composed of the two directional lines and a body line, the intersection of the boundaries is the combinatorial intersection of these lines. The intersections are either between two directional lines, for example, $(d_{ne})_A \bigcap (d_{nw})_B$

---

[2] The intersection of E_A with the other three areas N_B, S_B, E_B gives semi-infinite areas.

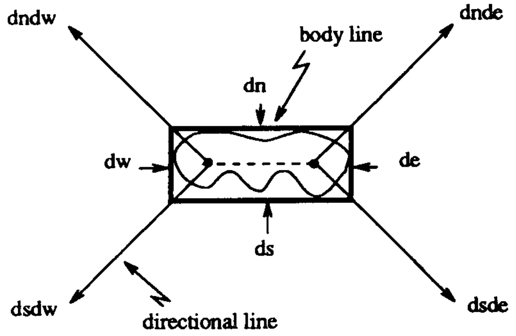


Figure 6: Body lines and Directional Lines of an object.

or between a direction line and body line, for example, $(d_e)_A \bigcap (d_{nw})_B$.

Generalizing the above argument, it can be concluded that,

- If two directional lines of the two bodies intersect such that $(d_{ij})_A \bigcap (d_{k\bar{j}})_B$ then the relation $(B\ [d_j]\ A)$ or $(A\ [\bar{d}_j]\ B)$ can be derived.
- Similarly, if $(d_{ij})_A \bigcap (d_{\bar{i}\bar{j}})_B$ then the relation $(B\ [d_{ij}]\ A)$ or $(A\ [d_{\bar{i}\bar{j}}]\ B)$ holds.
- If a body line of a body intersects with a directional line of another such that, $(d_{iA} \bigcap (d_{j\bar{i}})_B$, then the relation $(B\ [d_i]\ A)$ or $(A\ [d_{j\bar{i}}]\ B)$ can be derived.

Table 7 shows the possible directional relationships between two bodies A and B from the intersection of the components of the boundaries of their related directional areas. For example, from the intersection $NW_A \bigcap NE_B$, the relation $A\ [d_e]\ B$ can be derived while from the intersection $NE_A \bigcap NE_B$ no relationship can be derived.

The same advantages for adopting the *Intersection-based* approach apply as discussed in section 2, where the presented formalism can be compared with that proposed in [7]. The above formalism is also a natural extension to the *Intersection-based* approach which allows straightforward implementation of the formalism in systems adopting this approach.

4. EXTENDING THE INTERSECTION-BASED APPROACH FOR THE DEFINITION OF FLOW DIRECTION RELATIONSHIPS

B \ A	d_{NE}	d_{NW}	d_{SW}	d_{SE}	d_E	d_N	d_W	d_S
d_{NE}	-	d_E	d_{NE}	d_N	d_{NE}	d_{NE}	d_{NE}	d_{NE}
d_{NW}	d_W	-	d_N	d_{NW}	d_{NW}	d_{NW}	d_{NW}	d_{NW}
d_{SW}	d_{SW}	d_S	-	d_W	d_{SW}	d_{SW}	d_{SW}	d_{SW}
d_{SE}	d_S	d_{SE}	d_E	-	d_{SE}	d_{SE}	d_{SE}	d_{SE}
d_E	-	d_E	d_E	-	-	d_E	d_E	d_E
d_N	-	-	d_N	d_N	d_N	-	d_N	d_N
d_W	d_W	-	-	d_W	d_W	d_W	-	d_W
d_S	d_S	d_S	-	-	d_S	d_S	d_S	-

Figure 7: Possible directional relationships between two objects from the intersection of the direction and body lines associated with them. The elements in the table indicate the relation $A\ [R]\ B$.

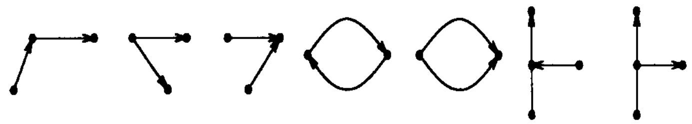
An important class of objects in geographic databases are those characterized by moving resources or carrying flow of a commodity from one location to another. These are usually represented by a linear spatial representation. Examples include segments of: roads, rivers, railways, power transmission lines, airline service routes, streams of drainage basins, utility lines (gas, water, sewage, telephone, etc).

Network analysis operations are used in GISs over such classes, for example, shortest route and resource allocation operations. Such operations are based on the spatial relationships between the basic components of such networks referred to as *flow-segments* in [1]. Topological spatial relationships between these *flow-segments* are the same as the relationships between two line objects with the constraint that flow continuity is maintained for flow analysis to be carried out. Thus there can be no flow analysis between two disjoint *flow-segments.*

In [2] the matrix representing relationships between two line objects is given which recognizes the boundary of a line as two disjoint points. The same matrix is used to represent the relationships between two flow-segments where the boundaries in this case represent the source $\delta_s A$ and destination $\delta_d A$ of the flow in flow-segment A as shown in figure 8(a). The possible relationships from the matrix are as shown in figure 8(b). The purpose of

$$\begin{bmatrix} \delta_s A \bigcap \delta_s B & \delta_s A \bigcap \delta_d B & \delta_s A \bigcap B^\circ \\ \delta_d A \bigcap \delta_s B & \delta_d A \bigcap \delta_d B & \delta_d A \bigcap B^\circ \\ A^\circ \bigcap \delta_s B & A^\circ \bigcap \delta_d B & A^\circ \bigcap B^\circ \end{bmatrix}$$

(a)



(b)

Figure 8: a) Intersection matrix for flow-direction relationships. (b) The possible flow direction relationships between two ***flow-segments***.

the exposition of the flow direction operations in the paper is to show how this important category of geographic database operations can be represented and related to the *Intersection-based* representation formalism.

CONCLUSIONS

In this paper two approaches to the development of representational formalisms for spatial relationships are identified, namely, ***interaction-based*** and *intersection-based* approaches. These are compared according to a set of identified properties of a representation formalism, namely, ***completeness***, ***soundness***, *uniqueness of representation* and *generality*. The intersection-based method is then used for the representation of cardinal directions between objects of arbitrary shapes and sizes.

There is a direct mapping between the intersection-based approach and its implementation in a database context which makes it a suitable formalism to be adopted for the derivation of qualitative spatial relationships in a GIS. However, extensions to the approach are still needed for the formalism to be *general* and to provide for *uniqueness of representation.*

ACKNOWLEDGMENTS

The work that resulted in this paper has been funded by the Science and Engineering Research Council through the IEATP programme, and their support is duly acknowledged. We would like to thank Dr. N.W. Paton for reading the original draft of the paper, and Dr. Keith G. Jeffery of Rutherford Appleton Laboratory, and Dr. J.M.P. Quinn representing ICL and Mr Neil Smith of Ordnance Survey for useful discussion on the subject of this paper.

REFERENCES

[1] Abdelmoty, A.I., Paton, N.W., Williams, M.H. Fernandes, A.A.A., Barja, M.L., Dinn, A. (1994) Geographic Data Handling in an Deductive Object-Oriented Database in *Proc. of the 5th International Conference on Database and Expert Systems Applications, DEXA '94*, Springer Verlag

[2] Abdelmoty, A.I., Williams, M.H., Paton, N.W. (1993) Deduction and Deductive Databases for Geographic Data Handling in *Design and Implementation of Large Spatial Databases, Third International Symposium, SSD '93*, eds. D. Abel, B.C. Ooi, Springer Verlag, **LNCS 692**, Singapore, p. 443-464

[3] Chang, S.K., Liu, S.H. (1984) Picture Indexing and Abstraction Techniques for Pictorial Databases, *IEEE Transaction on Pattern Analysis and Machine Intelligence*, **PAMI-6**, 4, p. 475-484

[4] Clarke, B.L. (1981) A Calculus of Individuals Based of 'Connection', *Notre Dame Journal of Formal Logic*, **22**, 3, p. 204-218

[5] Clementini, E., DiFelice, P., Oosterom, P.V. (1993) A Small Set of Formal Topological Relationships suitable for End -User Interaction in *Design and Implementation of Large Spatial Databases, Third International Symposium, SSD '93*, eds. D. Abel, B.C. Ooi, Springer Verlag, **LNCS 692**, Singapore, p. 277-295

[6] Cohn, A.G., Randell, D.A., Cui, Z. (1994) Taxonomies of Logically Defined Qualitative Spatial Relations in *Formal Ontology in Conceptual Analysis and Knowledge Representation*, eds. N. Guarino, R. Poli, Kluwer

[7] Cui, Z., Cohn, A.G., Randell, D.A. (1993) Qualitative and Topological Relationships in Spatial Databases in *Design and Implementation of Large Spatial Databases, Third International Symposium, SSD '93*, eds. D. Abel, B.C. Ooi, Springer Verlag, **LNCS 692**, Singapore, p. 396-315

[8] Egenhofer, M.J. (1989) A Formal Definition of Binary Topological Relationships in *Proc. 3rd International Conference on Foundations of Data Organization and Algorithms, FODO 89*, Paris

[9] Egenhofer, M.J. (1989) Spatial Query Language, *PhD thesis*

[10] Egenhofer, M.J. (1991) Reasoning About Binary Topological Relations in *Advances in Spatial Databases, 2nd Symposium, SSD'91*, eds. O. Gunther, H.J. Scheck, Springer Verlag, **LNCS 525**, Zurich, p. 143-161

[11] Egenhofer, M.J. (1993) Definitions of Line-Line Relations for Geographic Databases, *Data Engineering*, sbf 16, 3, p. 40-45

[12] Egenhofer, M.J., Franzosa, R.D. (1991) PointSet Topological Spatial Relations, *Int. J. Geographic Information Systems*, **5**, 2, p. 161-174

[13] Egenhofer, M.J., Herring, J.R. (1990) A Mathematical Framework for the Definition of Topological Relationships in *Proceedings of the 4th international Symposium on Spatial Data Handling*, **2**, p. 803-813

[14] Egenhofer, M.J., Sharma, J. (1993) Topological Relations between Regions in R^2 and Z^2 in *Design and Implementation of Large Spatial Databases, Third International Symposium, SSD '93*, eds. D. Abel, B.C. Ooi, Springer Verlag, **LNCS 692**, Singapore, p. ??

[15] Egenhofer, M.J., Clementini, E., Di Felice P. (1994) Topological Relations Between Regions With Holes, *Int. J. Geographic Information Systems*, **8**, 2, 129-142

[16] Evans, T.G. (1968) A Heuristic Program to Solve Geometric Analogy Problems, *Semantic Information Processing*, eds. M. Minsky, MIT press, Cambridge

[17] Frank, A.U. (1992) Qualitative Spatial Reasoning about Distances and Directions in Geographic Space in *Journal of Visual Languages and Computing*, **3**, Academic Press Limited, p. 343-371

[18] Franzosa, R.D., Egenhofer, M.J. (1992) Topological Spatial Relations Based on Components and Dimensions of Set Intersections, *Vision Geometry*, eds. R.A. Melter, A.Y. Wu, **SPIE - 1832**, p. 236-246

[19] Freksa, C. (1991) Qualitative Spatial Reasoning in *Cognitive and Linguistic Aspects of Geographic Space*, eds. D. Mark, A. Frank, Kluwer

[20] Freksa, C. (1991) Conceptual Neighborhood and its Role in Temporal and Spatial Reasoning, *Decision Support Systems and Qualitative Reasoning*, ed. M.G. Singh, L. Trave-Massuyes, Elsevier Science Publishers, p. 181-187

[21] Freksa, C. (1992) Using Orientation Information for Qualitative Spatial Reasoning in *Theories and Methods of Spatio-Temporal Reasoning in Geographic Space*, eds. A. Frank, I. Campari, Springer Verlag, p. 162-178

[22] Guesgen, H.W. (1989) Spatial Reasoning Based on Allen's Temporal Logic, *Technical Report TR-89-049*, Int. Computer Science Institute, Berkeley, California

[23] Güting, R.H. (1988) Geo-Relational Algebra: A Model and Query Language for Geometric Database Systems in *Advances in Database Technology- EDBT'88*, eds. J.W. Schmidt, M. Missikoff, Springer Verlag, p. 506-527

[24] Haar, R. (1976) Computational Models of Spatial Relations in *Technical Report, TR-478 MCS-72-0361-10*, Department of Computer Science, University of Maryland

[25] Hernandez, D. (1990) Using Comparative Relations to represent Spatial Knowledge in *Workshop RAUM*, eds. W. Hoeppner, p. 69-80

[26] Hernandez, D. (1993) Maintaining Qualitative Spatial Knowledge in *Spatial Information Theory, A Theoretical Basis for GIS : European Conference, COSIT'93*, eds. A.U.Frank, I.Campari, Springer Verlag, p. 36-53

[27] Herring, J. (1991) The Mathematical Modeling of Spatial and Non-Spatial information in Geographic Information Systems, *Cognitive and Linguistic Aspects of Geographic Space*, eds. D. Mark, A. Frank, Kluwer, p. 313-350

[28] Mukerjee, A., Joe, G. (1990) A Qualitative Model for Space in *Proceeding of the 8th National Conference on Artificial Intelligence, AAAI'90*, p. 721-727

[29] Peuquet, D.J., Ci-Xiang, Z. (1987) An Algorithm to Determine the Directional Relationship between Arbitrarily-Shaped Polygons in the Plane, *Pattern Recognition*, **20**, 1, p. 65-74

[30] Pullar, D.V., Egenhofer, M.J. (1988) Towards Formal Definition of Topological Relations Among Spatial Objects in *Proc. 3RD International Symposium on Spatial Data Handling*, p. 225-241

[31] Randell, D., Cui, Z., Cohn, A. (1992) A Spatial Logic Based on Regions and Connection in *Principles of Knowledge Representation and Reasoning, KR'92*, p. 165-176

[32] Smith, T.R., Park, K.K. (1992) Algebraic Approach to Spatial Reasoning, *International Journal of Geographic Information Systems*, **6**, 3, p. 177-192

[33] Vieu, L. (1993) A Logical Framework for Reasoning about Space, *Spatial Information Theory, A Theoretical Basis for GIS : European Conference, COSIT'93*, eds. A.U. Frank, I. Campari, Springer Verlag, p. 25-35

[34] Zimmerman, K. (1993) Enhancing Qualitative Spatial Reasoning - Combining Orientation and Distance, *Spatial Information Theory, A Theoretical Basis for GIS : European Conference, COSIT'93*, eds. A.U. Frank, I. Campari, Springer Verlag, p. 69-76

Cognitive Aspects of GIS

TOWARDS A COGNITIVE GIS

STEPHEN C. HIRTLE
Department of Information Science
University of Pittsburgh
Pittsburgh, PA 15260 USA

ABSTRACT. In this paper the relationship between cognition and GIS is discussed. It is argued that GIS can provide a useful metaphor for spatial cognition, which will lead to new questions. Such issues as multiple representations, scale versus resolution, and data integrity have natural correlates in a cognitive map. In turn, the examination of these issues will help guide the development of GIS. Finally, the role of spatial cognition, and of geography in general, for the exploration of other virtual spaces is considered.

1. INTRODUCTION

As geographic information systems become commonplace among researchers other than geographers, it becomes critical that they can represent space in an accessible and meaningful manner. Topological relations often provide the basis of queries to a geographic information system (GIS). One might be looking for a house 'near' work, one might be interested in finding an 'easy' route to follow, one might be interested in building a prison 'away' from a neighborhood, and so on. How can space be represented for the user and presented to the user so that an interaction is meaningful? Kuhn has argued for examining the underlying metaphors used in building a GIS interface (Kuhn, 1993). In this paper, I take a different approach and argue that cognitive psychology can benefit from looking closely at the underlying GIS data structures and assumptions, and, in turn, the results of such study will help guide the development of a cognitive GIS.

2. BACKGROUND

Research on cognitive maps that are used to organize space mentally has flourished in the past few years (see Golledge and Gärling, 1989; Hirtle & Heidorn, 1992, for reviews). The problem of determining the nature of the mental representation of physical space is necessarily complex, as the representation must integrate spatial information, such as location, orientation, distance, and density, with nonspatial information, such as semantic labeling, affective qualities, and cluster membership. Cognitive psychologists, myself included, tend to take a traditional approach of isolating single variables for study. This paradigm has proven effective, but as I argue in this paper, it is time for new paradigms and metaphors to be explored. In turn, the particular approach advocated has both immediate and long-term consequences for the future design of GIS modules and interfaces.

In reviewing the literature to ask "What is a cognitive map?", it became clear that there are very few existing metaphors for spatial representations. The most common is the rubber sheet model, which assumes that space is both continuous and close to two-dimensional, but that has been distorted, stretched, and pulled from the real space that it represents. Other partial metaphors that have been considered are strip maps for route information, topological models, propositional models, and 2-D projections of 3-D space.

Against this sparse theoretical background is a growing number of empirical observations, most of which are inconsistent with the metaphors above. These include, but are not limited to, hierarchies, alignments, rotations, and reference points. An example of hierarchies came about in a study by Michael Mascolo and myself, where we showed that memory for points on a screen could be biased by the labels attached to the points (Hirtle & Mascolo, 1986). Subjects were presented a display of labeled points, where labels came from one of two semantic clusters (government or recreation locations). A few key points were equidistant between members of the two classes. Despite repeated feedback to correct errors and various training procedures, we found that distances between points from the same conceptual cluster (or neighborhood) were under-estimated, whereas distances between points from two separate clusters were over-estimated. McNamara, Hardy, and Hirtle (1989) found similar clustering effects in a space with no pre-defined clusters. In this study, the distribution of points and the labels were chosen to be as homogeneous as possible, yet subjects still imposed srong hierarchical divisions on the space. The need to segment space makes many undifferentiated maps, such as topographical maps or many GIS displays, difficult to use and retain information without great study or practice.

This and similar studies focused on other structural components has lead Barbara Tversky (1993) to describe the representation as a cognitive collage rather than a cognitive map. By this, she has suggested that spatial knowledge is stored in differing formats, from multiple sources and differing points view. Collections of partial knowledge form thematic overlays of multimedia and come together to create our spatial encyclopedia.

In this paper, I would like to extend Tversky's (1993) arguments and suggest that additional metaphors are needed to understand spatial cognition. As recently argued, such metaphors are useful, not because they explain the spatial cognition, but because the suggest useful experiments to perform (Hirtle, 1994a). For example, if we are using the rubber sheet model for spatial cognition, then our experiments are designed to find the distortions. If we assume a topological/Euclidean continuum, then experiments are designed to test these aspects of space. Thus, additional metaphors are of use to the extent that generate useful and interesting research questions.

3. A METAPHOR OF GIS

One obvious metaphor that has been virtually ignored by cognitive psychologists is that of geographic information systems (GIS). Certainly I do not wish to imply that our minds are a miniature ARC/INFO, but the development of GIS has made explicit many storage and processing aspects, which might generalize to the study of spatial cognition. A partial list of potential factors includes

- raster versus vector approaches
- data integrity
- scale versus resolution
- overlays or multiple representations

Each of these points will be address in detail below.

3.1 Raster versus Vector Approaches

While the raster versus vector debate of how to best to store data in GIS is well-known within geography (e.g., Goodchild, 1992), there has been little discussion within cognitive psychology of the implication of these two alternative data structures. In a recent paper, Couclelis (1992) has argued that the technical question of the most appropriate data structure begs the philosophical question of the most appropriate conceptualization of geographic space." She argues that humans use both the object and field views in everyday reasoning, but for very different purposes.

Thus, the technical database question begs the cognitive map question of what is the appropriate conceptualization of cognitive maps, or alternatively when are the object and field views appropriate. Many empirical studies take an implicit view by studying a limited number of geographical objects, such as landmarks or regions, without addressing the underlying philosophical question that Couclelis (1992) proposes. Here, I consider two related issues that emerge from the consideration of the data storage format. The first is to consider the difference between point spaces and regions, which has received some attention in the cognitive literature. The second is to consider the difference between object and field views. While the latter is more closely related to the vector/raster debate, both arguments emerge when considering how to best store data.

Consider first the difference between point and regions. The question "How far is it from Chicago to New York?" takes Chicago to be a point, but the question "How long does it take to drive across Chicago?" takes Chicago to be a region (see Mark & Egenhofer, 1993, for further discussion of the term 'across'). However, other examples are ambiguous in their interpretation. For example, in the question "Does the fastest route from Indianapolis to Madison pass through Chicago?," either interpretation is possible for Chicago. The interpretation chosen would most likely depend on the context, individual preferences, and personal histories. How everyday objects viewed by user is an open question for further research.

Goodchild (1992) describes more fundamental differences implied by the raster/vector dichotomy, which suggests a distinction between well-defined boundaries and objects (the vector view) and a measurement of a feature or feature class over a regular tessellation of the plane (the raster view). The latter point of view suggests a sampling heuristic to test belongingness to a class (Goodchild, 1992). Under this view, the boundary of Colorado is best viewed under the vector model, but the extent of the Rocky Mountains is best viewed under a raster interpretation, as there is no clear boundary to the region. Returning to the questions listed above, if the term "Chicago" is taken to mean the Chicago city limits, which is a well-defined boundary, then we conclude subjects are using an object view for processing the information. However, if the term "Chicago" is taken to be the downtown area, then the boundary is not well-defined and the field view is appropriate.

The view of space is important not only as a cognitive issue, but also as a social issue. For example, the problem alluded to earlier of where to locate a prison is directly related to the perception of neighborhoods by planners and citizens. The "not-in-my-back-yard" response is likely to occur if the public and planners are using fundamentally different spatial models, or if they are using the same model, but with different calibrations.

3.2 Data Integrity

A second example of how GIS could provide a useful metaphor for the study of cognitive maps is the area of data integrity. How do the various cognitive systems (perceptual, tactile, semantic, etc.) pass knowledge? Does the transformation from linguistic to visual to spatial forms of knowledge alter the accuracy of spatial knowledge. If information is stored using both object and field views, depending on the nature of the information, then how is the information combined? Where are the "errors"? How can we find those errors?

Goodchild (1989, 1992) and Chrisman (1991) have enumerated a number of different types of errors in spatial data and GIS. There are not only multiple sources of error in measuring spatial data, but there are also fundamentally different kinds of error dependent on the underlying data model. For example, Goodchild (1992) notes that a statistical theory of error is relatively easy to include in a field view, yet all but impossible to include in object view. However, very little cognitive research has been done on measuring field representations.

One such study that has taken a field view for the study of spatial memory to isolate bias to certain processing stages was that of Huttenlocher, Hedges, and Duncan (1991). They describe a series of four experiments in which subjects try to reproduce, from memory, the location of a single dot in circle. Subjects were shown to recall the location of the dot in a circle, systematically shifted towards the center of the quadrant in which the point fell, where the horizontal and vertical axes form the imaginary boundarys of the quadrants. The empircal results are accounted for by a model that assumes an unbiased memory for the location, which is truncated by the boundaries at the time of response. Thus, the study by Huttenlocher, Hedges, and Duncan (1991) makes the strong claim that a bias in response need not imply that underlying cognitive representation is also biased. Within a GIS, similar distortions may occur during the output process, such as when forced to assign a point to a single cell of raster grid, even if the underlying representation suggests a greater accuracy or grain.

3.3 Scale versus Resolution

The third example to examine is one of scale vs. resolution. In traditional maps, these two tend to be correlated. Maps drawn for large areas include only major roads and major towns. As one zooms in on an area, the cartographer adds more detail, thus keeping the paper to ink ratio (Tufte, 1983) almost constant. However, since the data representation and the data presentation are independent, scale and resolution can be disambiguated in a GIS system. Stone, Fishkin, and Bier (1994) present one such implementation of this principle called a "magic lens." The lens is a movable filter that can provide enhanced detail to certain regions of a display, as seen in Figure 1. A fish-eye view is another similar technique to provide differential detail (Furnas, 1986; Sakar & Brown, 1992).

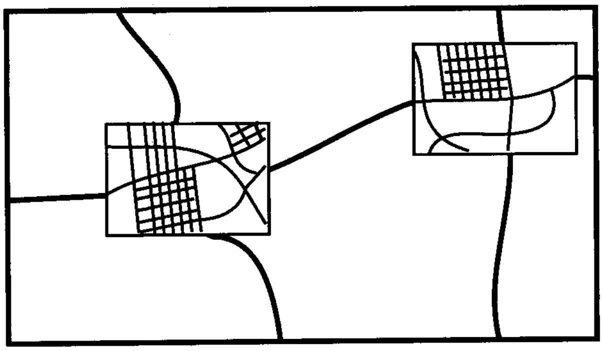


Figure 1. *Lenses showing selective detail, adapted from Stone, et al. (1994).*

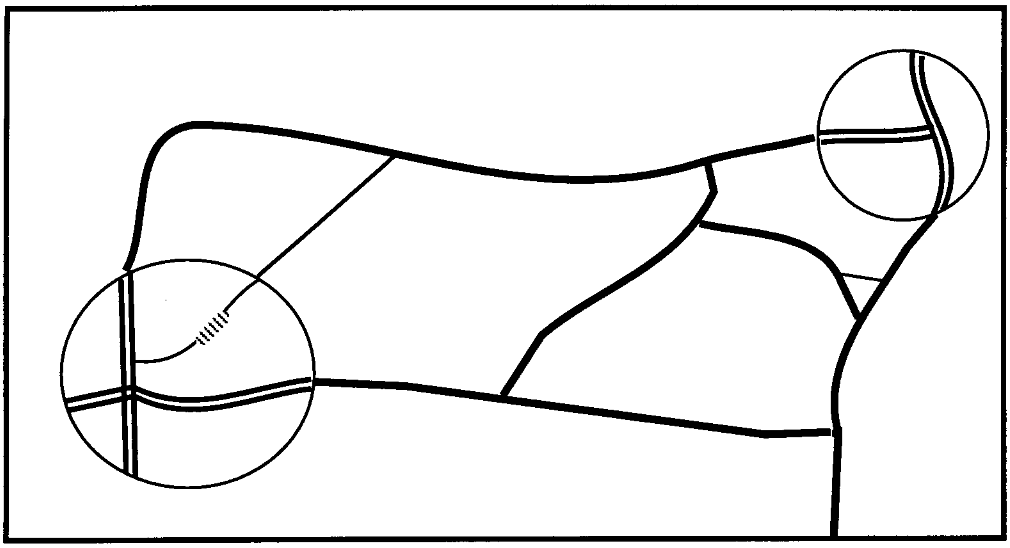


Figure 2. *Metric and topological information together in a single display. Metric information is shown in circles; topological information is shown in between the circles.*

The idea of differential scale and resolution plots is particularly intriguing for the modeling of route planning tasks (Car & Frank, 1994; Timpf, Volta, Pollock, & Egenhofer, 1992). For example, imagine the task of planning a route from Pittsburgh to Boston. There are at least two major tasks that need to be carried out: (1) planning a major interstate route between the two cities, which might entail traveling local connector roads if the interstate network is not complete and (2) planning the local connections to the interstate route in the start and terminal city. If these tasks can occur independently or even serially, there is little problem of scale and resolution. The first task would use a typical interstate map and not be concerned with local streets, whereas the second task would use a local map and not be concerned with the interstate system.

However, in many real-world planning problems, these two tasks are not independent. The choice of interstate may depend on the easiest highway to approach locally. If there are several major routes between the cities, then the choice of major route might depend on interaction of the local and global constraints. Thus, it then becomes impossible to solve the problem using a single scale and resolution. Instead, a combination of detailed local information and fuzzy global information is needed. Within a GIS system, this could be realized using either a modification of the Stone, et al (1994) approach, where the scale varies in the lens, not just the amount of detail, or a hierarchical model as shown in Figure 2. Here, the material in the circles is on a true Euclidean metric scale, but the information in between the circles is only topological. The network route chosen to get from the lower left circle to upper right circle may depend the perception of difficulty traveling on local roads, including a busy local road, as marked by in the lower left circle of Figure 2.

3.4 *Multiple Representations*

In GIS, information is often stored in separate layers, to be joined into a single representation during later stages of analysis. The cognitive question that this approach begs is whether or not there are many separate underlying representations, which could be accessed independently or together. These representations might exist for purposes of scale, as suggested in the previous section or separate representations may co-exist at the same scale. If we consider reference points to be important organizing structures, different problems may evoke different spatial landmarks. Thus, one's driving, walking, and jogging maps of a city will have different constraints. Furthermore, a student's perception of buildings to attend classes, shops to buy clothes or books, and for restaurants to eat lunch, may each evoke different layers of the cognitive atlas.

If each context resulted only in the stretching or shrinking of one's mental ruler, it would be of only limited interest. But given what we know of the structure of spatial representations, this is unlikely. Instead, each layer would have its own clusters, its own hierarchies, its own distortions, and so on. More importantly, this approach can predict the distortions in orientation judgments by looking at each layer separately.

Unlike the GIS, we cannot physically separate layers. Therefore, alternative methods must be developed. In my lab, I have taken two approaches to this question. One is to have subjects study a multi-attribute space, then test on specific aspects of the space. The other is opposite -- to study independent attributes then test the locations together. You could think of this as pulling apart the layers or putting the layers together.

In related work, Golledge (1992) asks to what extent do spatial inferencing tasks use the common functions of GIS, such as overlay, compression, spatial distributions, adjacency and connectivity, orientation and direction, spatial sequencing and ordering, among others. In further research, Golledge, Bell, and Dougherty (1994) have presented experimental evidence showing the difficulty that human have in overlaying information from discrete sources.

4. EXTENSIONS TO VIRTUAL SPACES

The study of spatial metaphors is not only important for the study of geographic spaces, but also for virtual systems and complex hyperspaces. In a recent paper (Hirtle, 1994b), I argued for traditional information specialists to look at the role of spatial information for the display, searching, categorization, and storage of data. Consider, for example, the problem of getting lost when searching a large hypertext system, such as mosaic (see Kim & Hirtle, in press, for a review). This "lost-in-hyperspace" phenomenon occurs for several reasons. First, real space has real constraints, whereas hyperspace does not. Nodes might join in a strict linear order, a tree, a network, a cycle or any number of other topologies. Some topologies are indicative of a book, others of a museum, and others of an unorganized wilderness. You-are-here maps are either absent or uninformative when present.

One solution to this problem is to turn to research in cognitive science to understand why people get lost in physical environments (Dillion, McKnight, & Richardson, 1990; Kim & Hirtle, in press). Research has demonstrated a number of heuristics that people use to understand space. There are different types of spatial knowledge, such as route and survey knowledge. In simple spaces, individuals begin to acquire survey knowledge upon the first exposure to the space, whereas in complex spaces, such as hospitals, survey knowledge is rarely acquired even after years of experience.

Furthermore, characteristics of the physical space affect the ability to navigate withing an environment. For example, architects and urban planners have learned that undifferentiated spaces are harder to learn than rich environments. Even an idea as simple as using different colors on different levels of a parking garage will increase the likelihood of recalling where your car was parked upon return. Thus, aids in helping the user structure space and differentiate neighborhoods should lead to fewer errors and greater satisfaction with hypertext systems (Kim & Hirtle, in press).

A second solution is to consider the metaphors that users adopt in hyperspace (Gray, 1990; Kim & Hirtle, in press). Here the focus is on the relationship between the virtual space and the users' understanding of the virtual space. A critical observation is that the virtual space need not have a physical correlate to be easily traversed, and the inclusion of a physical correlate does not guarantee avoiding disorientation. For example, understanding the mapping of a video game that assigns the top row to the bottom row, and the left edge to the right edge is easily understood and visualized, even if it is physically impossible in real-space. Likewise, people may find themselves lost in a museum of interconnected rooms, so the corresponding hyperworld with the same connected relations as the mseum rooms would be equally disorienting (Foss, 1989).

Disorientation is often the result of a lack of an appropriate metaphor, or the user adopting the incorrect metaphor. On-line aids, such as history trees, maps, and fish-eye views, can assist the user both in developing an appropriate metaphor and locating one's self in the virtual space. Pointers with some degree of redundancy will tend to more useful than pointers without redundancy. However, the exact methods which prove to be of the most use in a given situation will depend on the structure of the virtual space and the preferences of the user.

5. SUMMARY

In summary, in this paper I have attempted to outline the limitations of simple metaphors for cognitive maps, such as the rubber sheet model, and experiments designed to isolate single factors. Additional theoretical mileage can be gained by adopting richer metaphors, such as GIS, for the study of spatial cognition. An analysis, such as the one presented on scale versus resolution, is a natural extension of the using GIS as a metaphor for the study of cognition. Furthermore, there are direct extensions to the navigation and exploration of hyperspace and virtual worlds. It is my hope that future research in the area continues to refine many of the issues discussed today.

ACKNOWLEDGMENTS

The author wishes to thank the National Center for Geographic Information and Analysis at the University of Maine and the National Science Foundation for their support through the NCGIA Visiting Scientist Program. The author is also grateful to Max Egenhofer, Andrew Frank, Scott Freundschuh and David Mark for their comments concerning the issues presented in this paper.

BIBLIOGRAPHY

Car, A., & Frank, A. (1994). Modeling of the hierarchy of space applied to large road networks. Paper presented at IGIS'94, Monte Verita, Ascona, Switzerland.

Chrisman, N. R. (1991). The error component in spatial data. In D. J. Maquire, M. F. Goodchild, & D. W. Rhind (Eds.), *Geographical information systems*. Essex, England: Longman.

Couclelis, H. (1992). People manipulate objects (but cultivate fields): Beyond the raster-vector debate in GIS. In A. U. Frank, I. Campari, & U. Formentini (Eds.), *Theories and methods of spatio-temporal reasoning in geographic space.* Berlin: Springer-Verlag.

Dillion, A., McKnight, C., & Richardosn, J. (1990). Navigation in hypertext: A critical review of the concept, in D. Diaper (Ed.), *Human-Computer Interaction - INTERACT '90* (Amsterdam: Elsevier Science Publishers).

Foss, Carolyn L.(1989). "Detecting Users Lost: Empirical Studies on Browsing Hypertext". Technical Report (Rapports de recherche) No 972, INRIA, Sophia-Antipolis, February 1989.

Furnas, G. W. (1986) Generalized fisheye views. In *Human Factors in Computing Systems, CHI `86.* (pp. 16-23), Boston.

Golledge, R. G. (1992). Do people understand spatial concepts: The case of first-order primitives. In A. U. Frank, I. Campari, & U. Formentini (Eds.), *Theories and methods of spatio-temporal reasoning in geographic space.* (pp. 1-21). Berlin: Springer-Verlag.

Golledge, R. G. , Bell, S., & Doughtery, V. J. (1994). Reasoning and inference in spatial knowledge acquisition: The cognitive maps as an internalize GIS. Paper presented at the Association of American Geographers Meeting, March 29 - April 4, San Francisco.

Golledge, R. G., Gärling, T. (1989). Environmental perception and cognition. In E. H. Zube and G. T. Moore (Eds.), Advances in environment, behavior, and design, Volume 2 (pp. 203-236). Plenum.

Goodchild, M. F. (1992). Geographical data modeling. *Computers & Geosciences, 18,* 401-408.

Goodchild, M. F. (1989). Modeling error in objects and fields, in M. F. Goodchild, & S. Gopal (Eds.), *Accuracy of spatial databases.* London: Taylor & Francis.

Gray, S. (1990). Using protocol analyses and drawings to study mental model construction during hypertext navigation. *International Journal of Human-Computer Interaction, 2,* 359-377.

Hirtle, S. C. (1994a). The cognitive atlas: Using GIS as a metaphor for memory. To appear in Golledge, R. G., & Egenhofer, M. J. *Spatial-temporal reasoning.*

Hirtle, S. C. (1994b). Spatial information science. Paper to be presented at the Fourth International Symposium on Information Science, November 2 - 4, 1994, University of Graz, Austria.

Hirtle, S. C., Heidorn, P. B. (1992). The structure of cognitive maps: Representation and processes. In T. Gärling, & R. G. Golledge (Eds.), Behavior and environment: Psychological and geographical approaches.

Hirtle. S. C., Mascolo, M. F. (1986). The effect of semantic clustering on the memory of spatial locations. Journal of Experimental Psychology: Learning, Memory and Cognition, 12, 181-189.

Huttenlocher, J., Hedges, L. V., & Duncan, S. (1991). Categories and particulars: Prototype effects in estimating spatial location. *Psychological Review, 98,* 352-376.

Kim, H., & Hirtle, S. C. (In press). Spatial metaphors and disorientation in hypertext browsing. To appear in *Behavio;r & Information Technology.*

Kuhn, W. (1993). Metaphors create theories for users. In A. U. Frank, & I. Campari (Eds.) *Spatial information theory: A theoretical basis for GIS* (pp. 336-376). Berlin: Springer-Verlag.

Mark, D. M., & Egenhofer, M. J. (In press). Modeling spatial relations between lines and regions: Combining formal mathematical models and human subjects testing. *Cartography and Geographic Information Systems, 21.*

McNamara, T. P., Hardy, J. K. and Hirtle, S. C. (1989) Subjective hierarchies in spatial memory. *Journal of Experimental Psychology: Learning, Memory, and Cognition, 15,* 211-227.

Sakar, M., & Brown, M. H. (1992). Graphical fisheye views of graphs. In Proceedings of ACM CHI'92, (pp. 83-91). Monterey, CA.

Stone, M. C., Fishkin, K., & Bier, E. A. (1994). The movable filter as a user interface tool. In the proceedings of the ACM CHI '94 Conference on Human Factors in Computing Systems, (pp. 306-312).

Timpf, S., Volta, G. S., Pollock, D. W., & Egenhofer, M. J. (1992). A conceptual model of wayfinding using multiple levels of abstraction. In A. U. Frank, I. Campari, & U. Formentini (Eds.), *Theories and methods of spatio-temporal reasoning in geographic space.* (pp. 348-367). Berlin: Springer-Verlag..

Tversky, B. (1993). Cognitive maps, cognitive collages, and spatial mental models. In A. U. Frank, & I. Campari (Eds.), *Spatial Information theory: A theoretical basis for GIS.* Berlin: Springer-Verlag.

Tufte, E. R. (1983). *The visual display of quantitative information.* Cheshire, Conn.: Graphics Press.

HANDLING SPATIO-TEMPORAL UNCERTAINTIES OF GEO-OBJECTS FOR DYNAMIC UPDATE OF GIS DATABASES FROM MULTI-SOURCE DATA

Ryosuke SHIBASAKI
Institute of Industrial Science, The University of Tokyo
7-22-1 Roppongi, Minato-ku, Tokyo 106 JAPAN

ABSTRACT:

As more GIS's are used in many institutions and fields, the importance of keeping GIS database current is keenly recognized. To fulfill the requirement, a concept of dynamic update of GIS databases using a wide variety of data from multi-sources and a methodological framework are proposed. At first, a model to represent geo-objects with spatio-temporal uncertainties is proposed. The model can serve as a basis to handle geo-objects with different levels of spatio-temporal uncertainties especially in integrating data from multi-sources. Secondly, a method is proposed to estimate or reconstruct status of relatively static geo-objects which only emerge and extinguish using observational data from multi-sources. Further extension of the method is required to estimate status of more dynamic geo-objects. To merge geometric data with different levels of uncertainties, a method to estimate "most probable" position of a line feature is proposed. It is necessary to validate the effectivity of this methodological framework using actual data of urban object such as houses and structures.

Keywords: Fuzziness, Data Modeling, Dynamic Update, Data Integration

1.BACKGROUND AND OBJECTIVES OF THE STUDY

As more GIS's are used in many institutions and fields, the importance of keeping GIS database current is keenly recognized. Especially in handling objects which change dynamically, it is crucially important to keep GIS databases "fresh". This requirement is found in many applications such as fixed property management for taxation, fire service and facility management and so forth.

To fulfill the requirement, conventional aerial survey and satellite remote sensing are not always helpful. Although aerial photographs and satellite imagery can cover wide areas simultaneously, they are "snapshots" and the frequency of observation is often limited, for example, because of weather condition and financial limitations. It may be often necessary and beneficial to rely upon non-survey data obtained from other sources in daily works. For example house inspection reports from fire service convey current information on construction/removal of houses and their structures. Construction permits and notice of removal of houses submitted to local governments are another example of data which can be potentially useful in updating GIS databases. These "fresh" data are potentially very useful to keep GIS databases current, although they do not fulfill spatial accuracy standards for mapping and some of newly (illegally)-built houses may not be captured by these data.

On the other hand, automated interpretation of aerial photos, which shows rapid advances in recent years, is very expecting to automate update of GIS databases. However, it is widely recognized that the information which can be obtained only from images are limited and that integration with other geographic data and knowledge is effective in improving the accuracy of automated interpretation.

In this paper, a process is called dynamic update, where GIS databases are kept current by integrating data, survey and non-survey data, from multi-sources.To realize dynamic update, the following major problems with existing GIS can be identified.

1) No sufficient methods to estimate or reconstruct status of geo-objects using data with different levels of accuracy and reliability.

2) In estimating status of geo-objects using data with different levels of accuracy and reliability, we have to handle geometric and thematic uncertainties of observational data explicitly. In addition, estimated status of geo-objects may include certain level of uncertainties. Existing GIS, however, provides no sufficient basis to represent and manage geo-information and geo-objects with uncertainties.

The objectives of this paper are:

(1) to propose a model to represent geo-objects with geometric and temporal uncertainties to provide a sound basis in handling them in GIS databases,

(2) and to develop a method to estimate status of geo-objects by integrating observational data with different levels of uncertainties.

2.CONCEPT OF DYNAMIC UPDATE OF GIS DATABASES

In this paper, dynamic update of GIS databases is defined as a process to update GIS databases continuously by integrating a variety of observational data from multi-sources ranging from automated aerial photo interpretation to documents submitted for construction permits (**Fig.2-1**). To estimate status of geo-objects efficiently and reliably, knowledge on behavior of geo-objects and on uncertainties and accuracy of observational data is indispensable. Examples of observational data are submitted address and drawings of planned house or development for construction permits, notice of removal of houses (which indicates house ID etc.), daily inspection reports by fire service conveying information on house construction and removal together with their structures, aerial photo interpretation (visual and automated) and design drawing of infrastructure such as roads.

Fig.2-2 shows factors characterizing observation methods. Accuracy and reliability in this figure mean, in case of automated aerial photo interpretation, probability of correct object-recognition and, in case of official notice such as construction permits and notice of removal, probability that events which must be notified are actually notified.

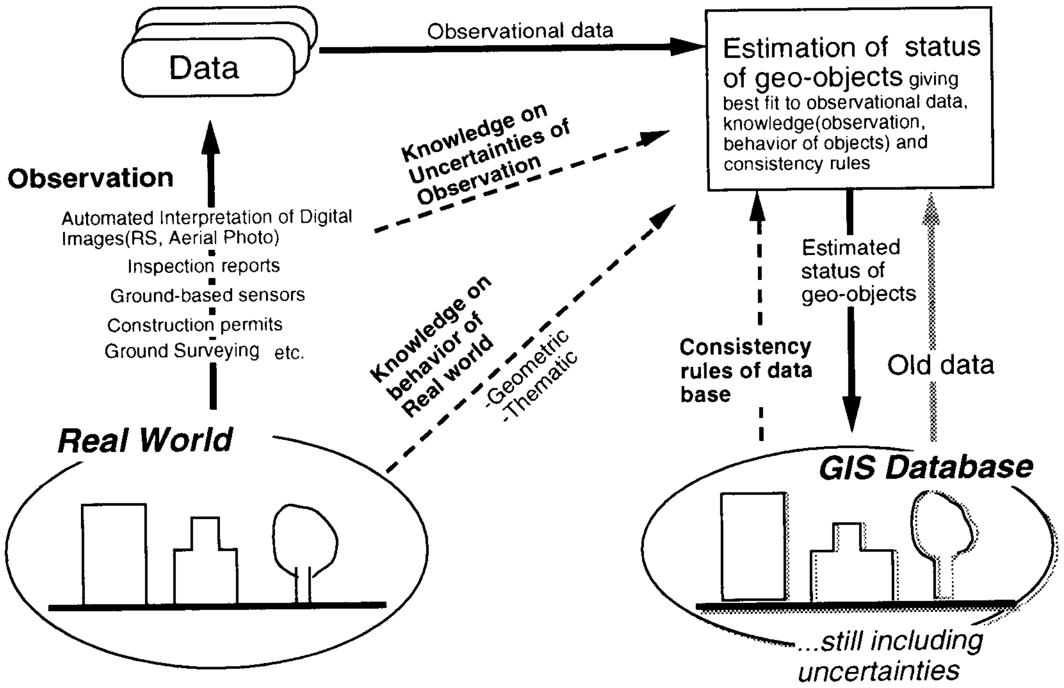


Fig.2-1 Dynamic Update of GIS Database
--------A Reconstruction Process of Real World using Observation Data from Multi-Sources

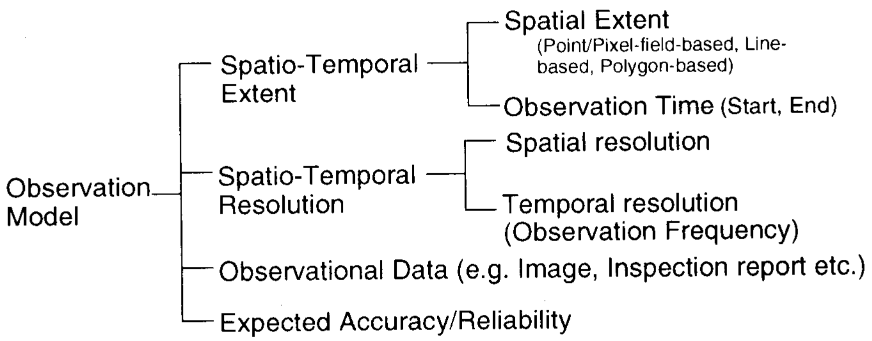


Fig.2-2 A Model of Observation

To realize dynamic update, it is necessary to handle geographic observational data with uncertainties. Geo-objects, which are estimated and represented in GIS databases, also include uncertainties. As a basis to realize dynamic update, a model to represent geo-objects with spatio-temporal uncertainties is keenly demanded.

3. A MODEL TO REPRESENT GEO-OBJECTS WITH SPATIO-TEMPORAL UNCERTAINTIES

3.1 Introduction

As shown in **Fig.3-1**, geo-objects are represented by "state" bounded by "events" in spatio-temporal space as proposed in (Langran,1992) . State includes both geometric information such as location and shape and thematic information. Geometric information is supposed to be represented by vector data with formal data structure (Molenaar, 1990). And it is assumed that a whole space is divided to polygons exclusively.

Geo-objects have constraint conditions on their dynamic behavior such as temporal pattern

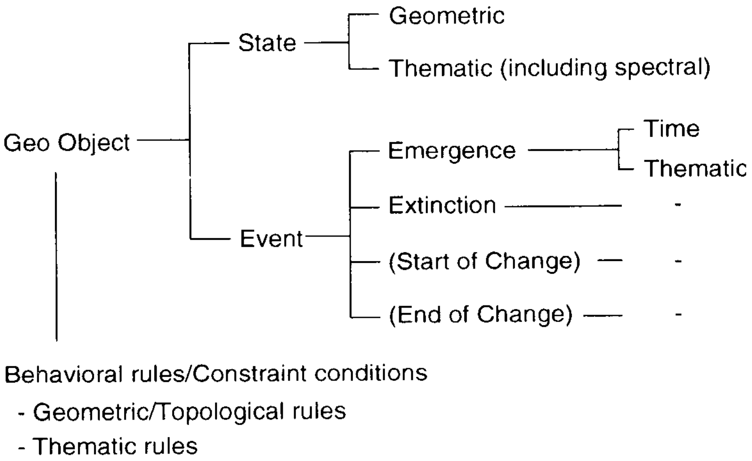


Fig.3-1 Definition of a Geo-Object

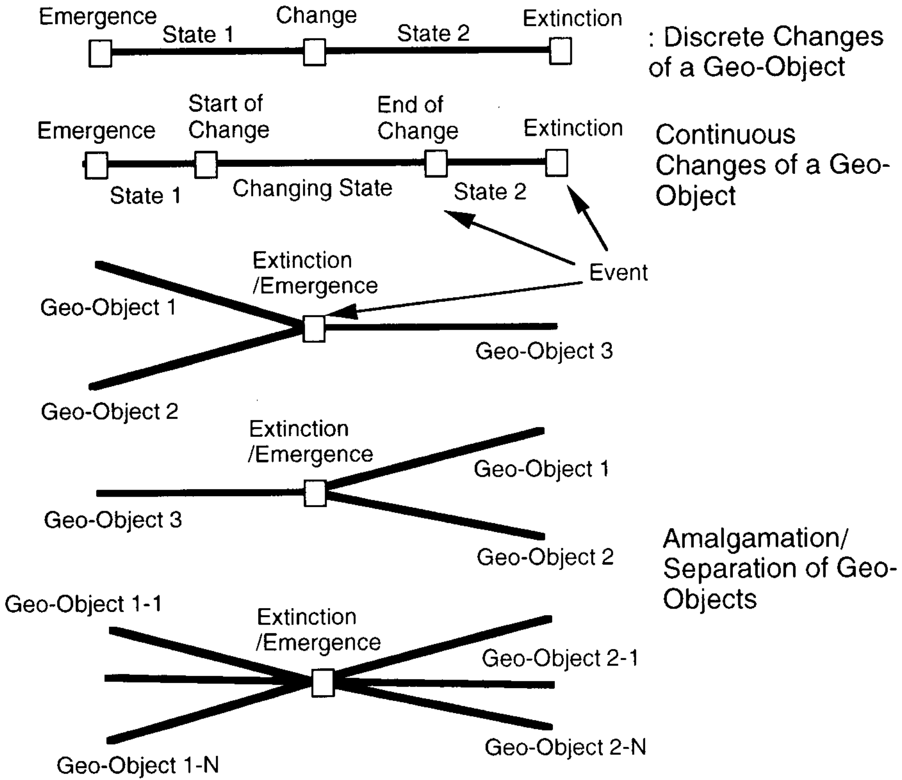


Fig. 3-2 Dynamics of Geo-Objects and Basic Components of Temporal Chains

of changes of state and probability event occurrence such as emergence/extinction. For example, houses have only emergence and extinction events except extension. And houses never overlap nor cross block boundaries. Probability of emergence and extinction can be estimated by conditions of land parcel, household and house itself such as age. These constraint conditions and knowledge are included in "behavioral model".

Events are emergence, extinction, discrete change and start and end of continuous changes as shown in **Fig.3-2**. Amalgamation and separation are natural connection of events. Temporal network where each event is recognized as a node connected by "state" links is called temporal chain in this paper. Under the assumption that object identity is based on geometric information such as shape of boundaries and that a whole space is divided to polygons exclusively, a geometric change (e.g.extinction) of one geo-object (polygon) causes extinction/ emergence of neighboring geo-objects (**Fig.3-3**).

In some cases, a number of geo-objects emerge as a result of "big" event such as large scale development projects. It is convenient to define a composite event as shown in **Fig.3-4**. A composite event have not only spatio-temporal extent as combination of each event of each object but also it's own thematic information such as "name of

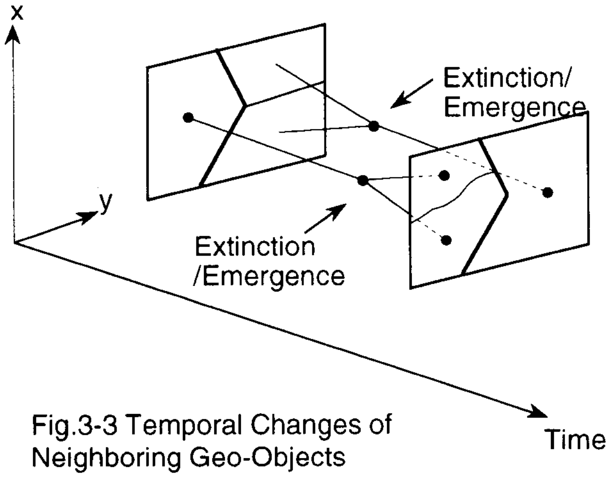


Fig.3-3 Temporal Changes of Neighboring Geo-Objects

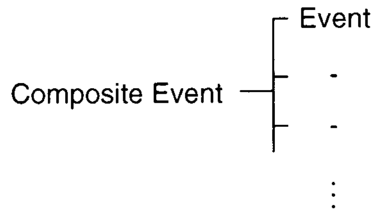


Fig. 3-4 A Composite Object and Event

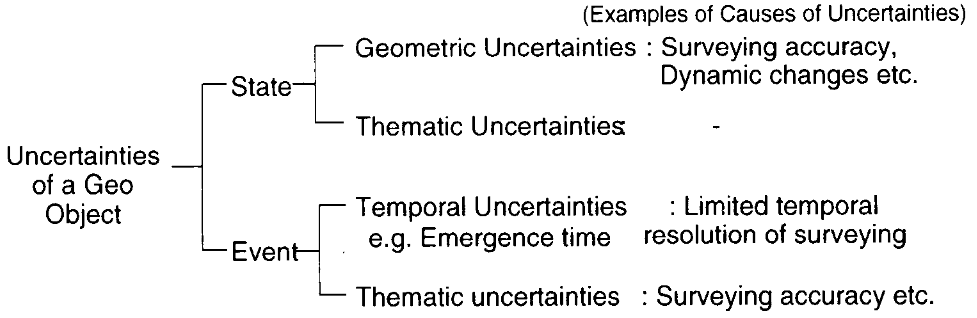


Fig.3-5 Uncertainties of Geo-Objects

development project".

Uncertainties are included in every aspect of geo-object definition as shown in **Fig.3-5**. The uncertainties consist of geometric uncertainty, temporal uncertainty and thematic uncertainty. This paper primarily focus geometric and temporal uncertainties .

3.2 A Model to Represent Geometric Uncertainties in GIS Databases

A model to represent geometric uncertainties is shown in **Fig.3-6**. In this model, parameters of characterizing uncertainty level such as a radius of possible location are attached to vector representation. This model can not only represent geometric uncertainties graphically but also serve as a basis of a database of geo-objects with geometric uncertainties, because basic spatial operations and a spatial addressing method can be developed based on the model. Basic spatial operations include computation of expectation and variance of distance between two

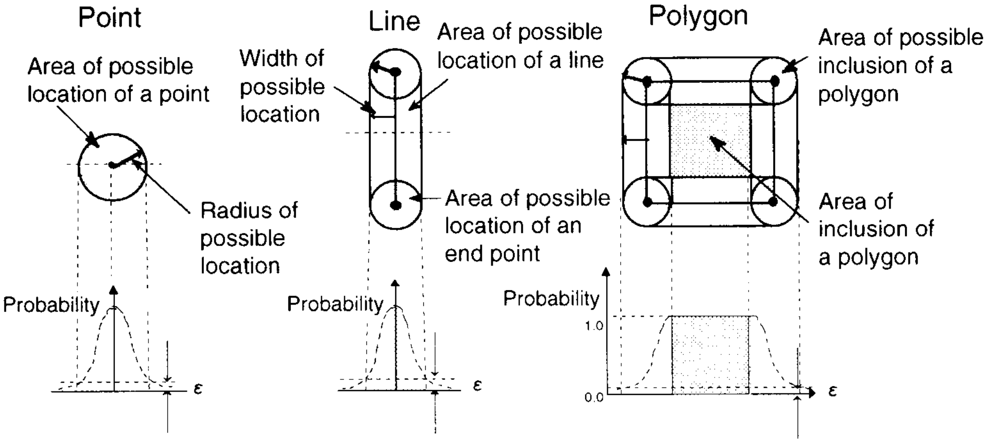


Fig.3-6 Representation of Geometric Uncertainties

Tab.3-1 Measures and Relations of Geometric Features with Positional Uncertainties

1) Measures

Line feature: length Polygon: area

2) Geometric relations

	Point	**Line**	**Polygon**
Point	Distance, Coincidence	Inclusion, Minimum distance, etc.	Inclusion, Minimum distance, etc.
Line	-	Intersection, Coincidence, Minimum distance, etc.	Intersection, Inclusion, Minimum distance, etc.
Polygon	-	-	Intersection, Inclusion, Boolean operation etc.

points and identification of possible and definite intersections between lines as well as a spatial addressing method are already developed (**Tab.3-1**,Shibasaki,1993). Positional uncertainties may causes uncertainties of topological relations as shown in **Fig.3-7**. Only topological relations without uncertainties are represented by a data structure because other relations with uncertainties can be identified by the spatial queries in **Tab.3-1**.

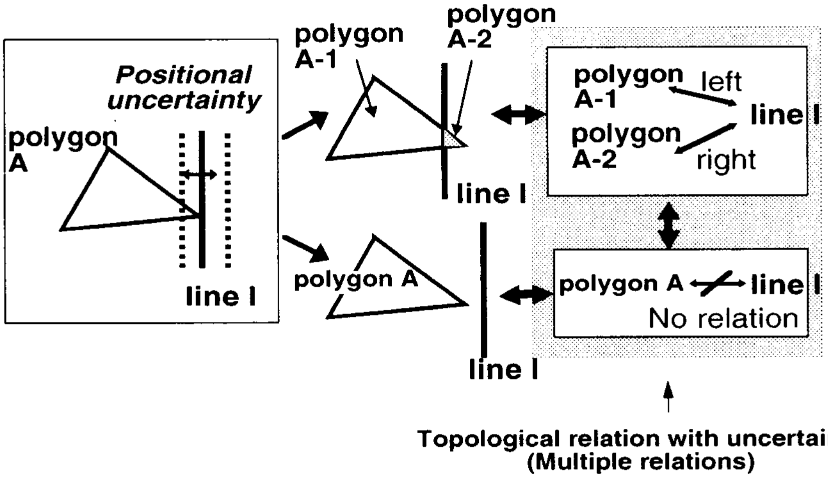


Fig.3-7 Uncertainties of Topological Relations due to Locational Uncertainties of Geometric Features

To estimate status of geo-objects with geometric uncertainties, it is useful to represent explicitly following topological conditions in a GIS database.

1) A point feature is in a polygon, or out of a polygon feature.
2) A point feature is not on a line feature.
3) A line feature does not cross nor overlap a line feature.
4) A line feature is included or not included in a polygon feature.

Other topological relations are represented by the format data structure.

3.3 A Method to Measure Geometric Uncertainties of Line Features

In integrating geographic information, levels of geometric uncertainties have to be assessed in advance. Conventionally, as for geometric uncertainties, only coordinate errors of point features are discussed. However, to estimate "most probable" position of line by integrating data from repeated or overlapped measurement, it is necessary to quantify the different levels of uncertainties of each measurement. **Fig.3-8** illustrates a concept of measuring positional discrepancies of line features. Here we assume that a measurement process of line position is a series of measurements with sufficiently small regular intervals of point locations along an actual line. "Virtual" measured points should be located on a measured line. Positional uncertainties of a line feature is assessed by error vectors connecting a point on

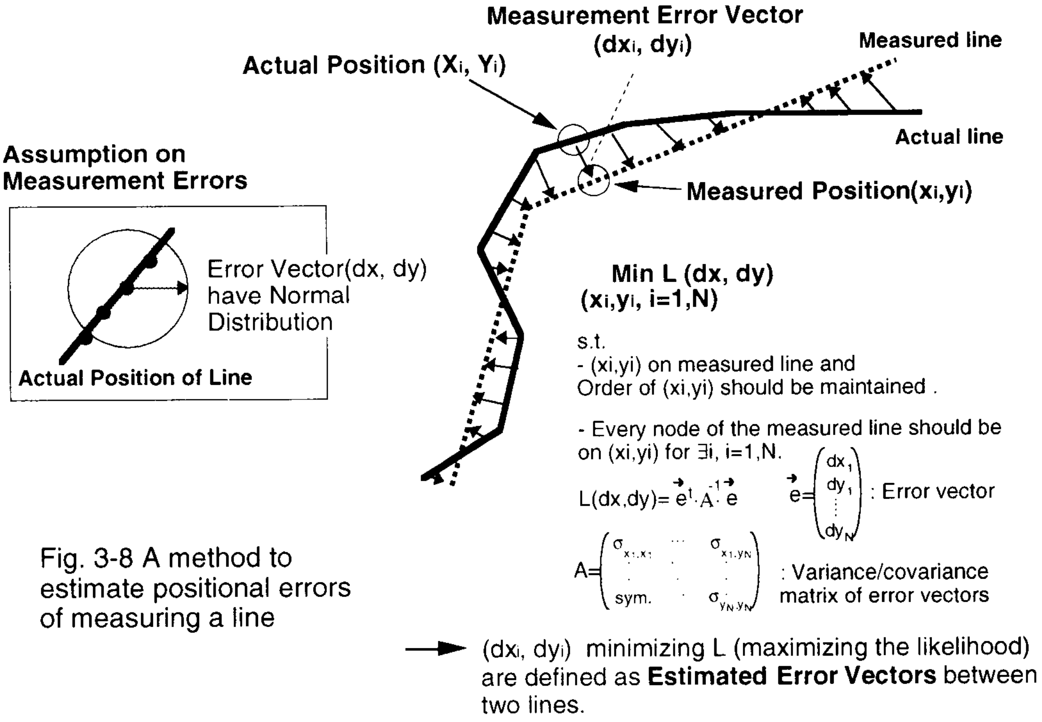


Fig. 3-8 A method to estimate positional errors of measuring a line

an actual line with a measured point on a measured line. Actually, the position of measured points on a measured line can vary on a measured line. By assuming that error vectors have multi-dimensional normal distribution, position of measured points on a measured line can be estimated by maximizing the likelihood of occurrence of error vectors under several natural constraint conditions such as every node of a measured line should be on destination of some of error vectors. The method can be applied to defining a discrepancy between two line features in 3D space.

3.4 A Model to Represent Temporal Uncertainties of Geo-Object

Temporal uncertainties of geo-object is represented by uncertainties of time, order and synchronization (e.g. amalgamation) of event occurrence. The uncertainty of occurrence time is defined by earliest time, latest time and expectation of time (**Fig.3-9**). A probability distribution function of occurrence time between the earliest time and the latest time can be an arbitrary curve. The

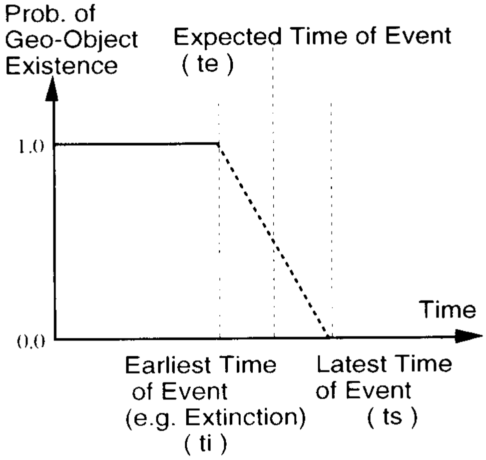


Fig. 3-9 Representation of Temporal Uncertainties of Events

earliest time and the latest time is defined as the earliest and latest time where accumulated probability of event occurrence exceed a threshold value respectively.

Fig.3-10 shows an example of uncertainties of time, order and synchronization of events. If only time of occurrence is uncertain, while the order and synchronization of events have no uncertainties, a network structure of the temporal chain which represents the order and synchronization of events have no uncertainties. Occurrence time of each event is represented by a set of earliest time and latest time which fulfill constraint conditions representing the consistent order and synchronization of events (consistent structure of the temporal network (Dechter et al 1991)(Console and Torasso,1993).

If there are uncertainties of order and/or synchronization of events, network structures have also uncertainties. In an example as shown in **Fig.3-11**, three objects, A,B and C are amalgamated into one object. There are several possibilities. One is amalgamation of A and B followed by amalgamation of C. The other is amalgamation of B and C followed by that of A. In such cases, only

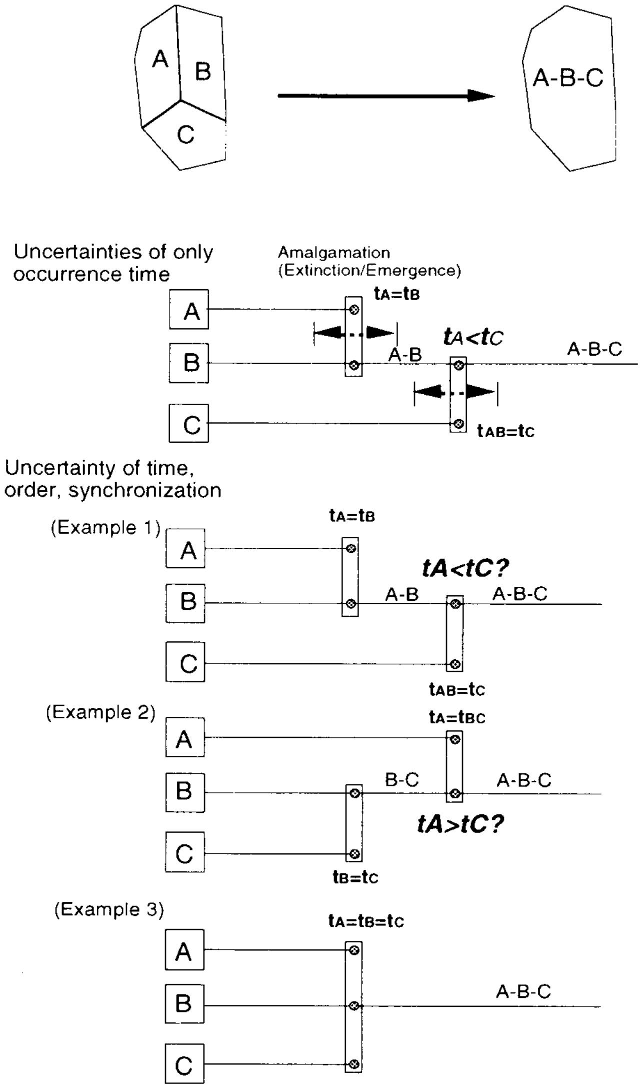


Fig. 3-10 Uncertainty of Occurrence Time, Order and Synchronization of Event Occurrence

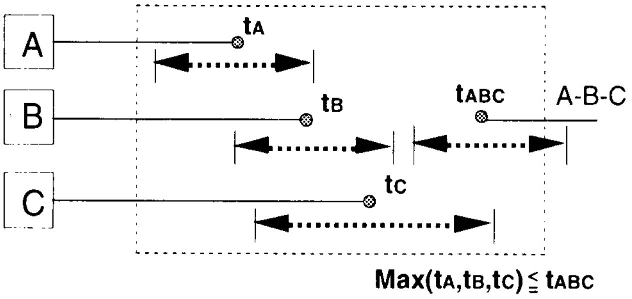


Fig. 3-11 Representations of Temporal Uncertainties in a Temporal Chain

time range of extinction of A,B and C and emergence of A-B-C are represented explicitly together with relevant consistency constraint conditions e.g.max.(t_A,t_B,t_C)< or = t_{ABC}.

In representing uncertainties in temporal network, following basic consistency constraint conditions have to be implemented simultaneously.

1) Emergence should occur before all other events of an object.

2) Extinction should occur after all other events of an object.

3) Start of Change should occur before End of Change of an object.

4) Extinction of a object due to changes of shape/position should induce extinction of at least one neighboring object and should be followed by emergence of at least one new object.

Constraint condition 4) comes from the assumptions that a whole space is divided into polygons exclusively and that objects are identified by shape and location . To identify the constraint condition 4), it is necessary to uncover overlapping polygons in different time-slices. Space-time composite is effective to improve efficiency in uncovering the overlapping polygons.

In addition to the above conditions, several conditions may be added such as "emergence of a object should occur at least 10 years later than extinction of a prior object at the same location" and "extinction of a object should occur before emergence of another object" (cause-effect relationship) and so forth.

3.5 Space-Time Composite

To generate a time-slice map of any time, it is necessary to identify (at least portion of) spatial topological relations between geometric features in different time slices. And to link objects in temporal chain which are overlapping with each other, spatial relations of the objects have to be identified.

If we assume that changes of geo-object may be caused by only emergence and extinction, it is easy to overlay geometric data of all geo-objects at any time onto one time-slice and to rebuild spatial topological relations by a method proposed by the author (Shibasaki, 1993) with the consideration of spatial uncertainties of the geometric data. This is a space-time composite (**Fig.3-12**) proposed by Langran (Langran, 1992). By building a space-time composite, spatial topological relations of geo-objects at any time slice can be easily retrieved.

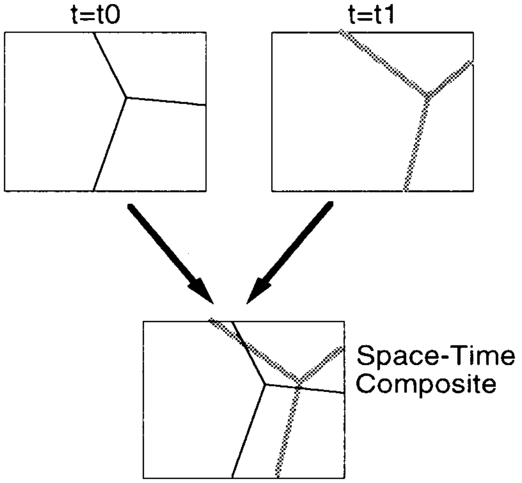


Fig.3-12 A Concept of Space-Time Composite

4. ESTIMATION OF STATUS OF GEO-OBJECTS USING DATA FROM MULTI-SOURCES

4.1 Estimation of Temporal Chains of Geo-objects

Fig.4-1 illustrates a process of estimating status of geo-objects using data from multi-sources. In this example, temporal changes of houses in urbanized areas is estimated for dynamic update of large-scale map databases. We assume that houses can only emerge or be removed. For relatively short time-span such as five year at most, possible temporal chains can be simple as shown in **Fig.4-1**. Posteriori probability of event occurrence for each type of hypotheses of temporal chains and time of occurrence can be computed using observational data (e.g.construction permits and aerial photographs etc.) by Bayesian theory. A type of hypothesis and time of occurrence which maximize posteriori probability is assumed to be "most probable" combination of type of temporal chain and time of occurrence (**Fig.4-2**). By setting a threshold value, earliest time and latest time of event occurrence are computed. Apriori probability of emergence and extinction(removal) of houses can be computed by conditions of a land parcel and a house. Reliability of observational data can also be estimated through comparison with actual changes.

Geometric (shape and location) data from interpretation of aerial photos are compared with "old" data by the method described in section 3.3. If the discrepancy is large, an object

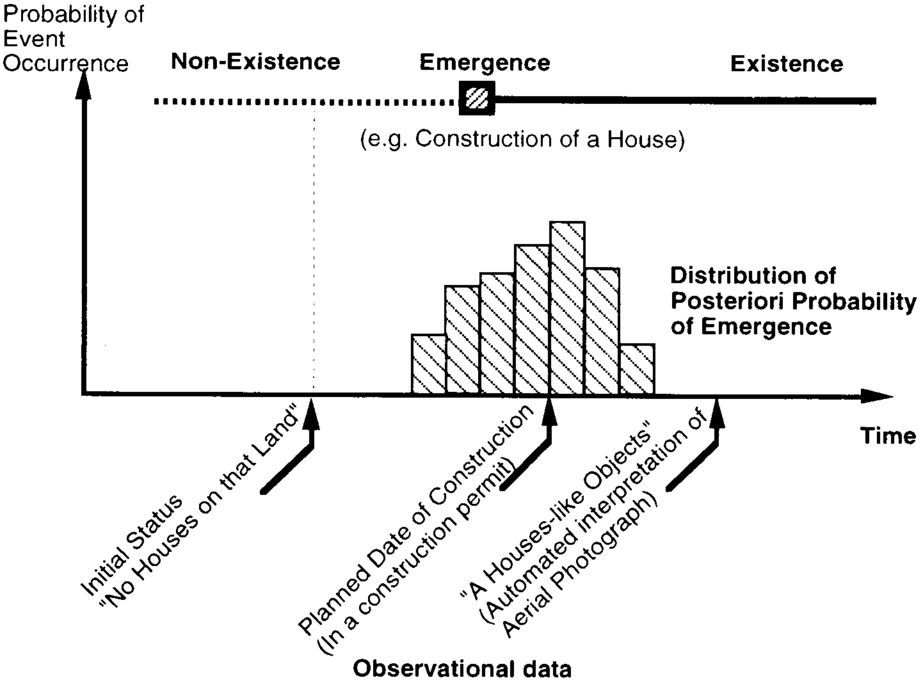


Fig.4-2 Estimation of Emergence Time using Bayesian Theory

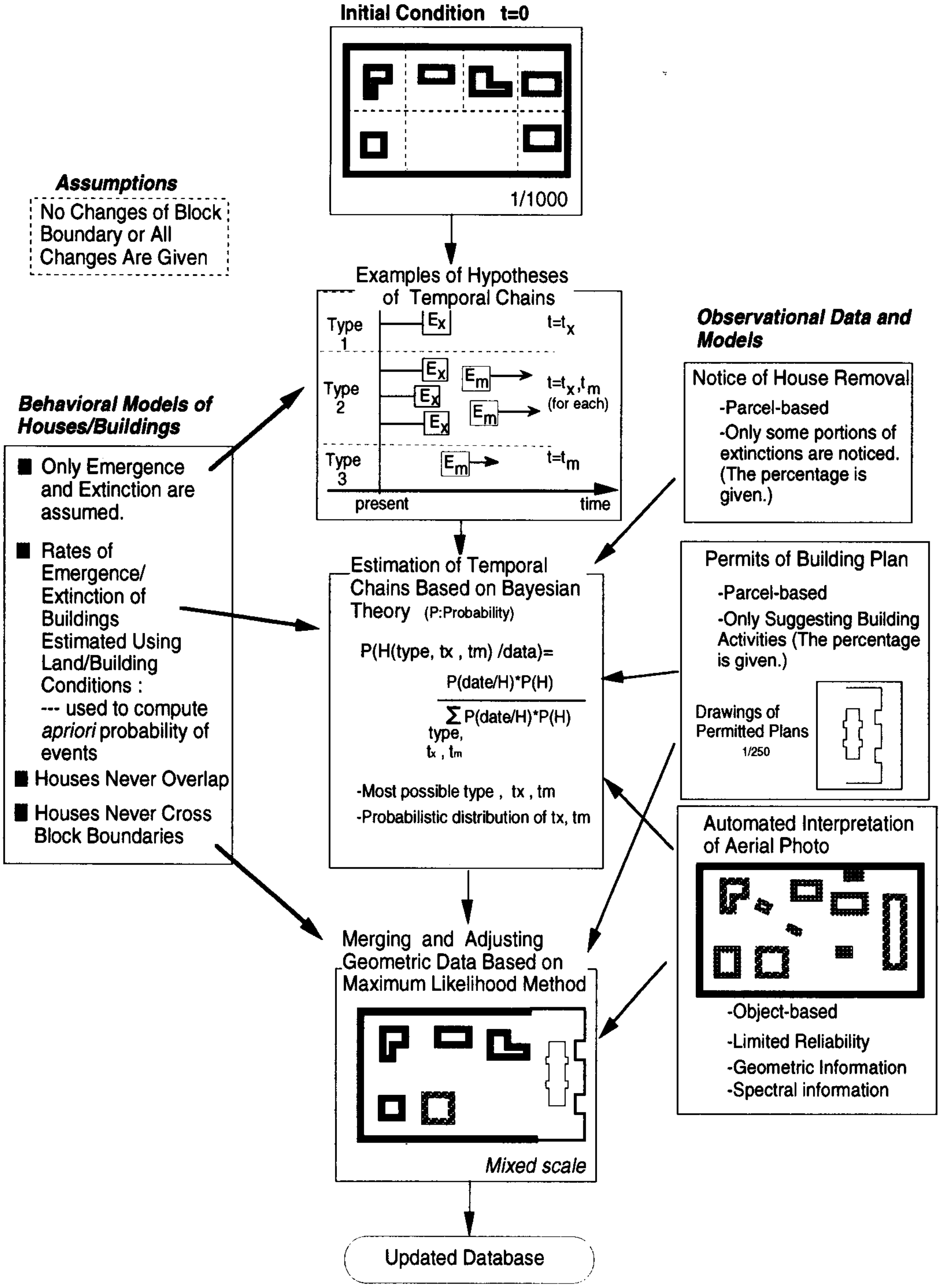


Fig.4-1 A Estimation/Reconstruction Method for Dynamic Update of GIS Databases Using Data from Multi-Sources

identified in aerial photo is recognized as a new one with some probability. Otherwise, it may be recognized as the identical house. In case the "old" data include albedo data, it can be effectively used for identification of object replacement. It is assumed that these probability of object identification can be given from experiments.

4.2 Estimation of Geometric Status of Geo-objects

After the estimation of temporal chains of houses, geometric data is updated. The followings should be taken into account in estimating geometric status of geo-objects.

1) In case measurement of geometric features, especially line features are repeated or overlapped, most probable position of a line feature have to be estimated. Examples of repeated measurement data are existing geometric data in a GIS database, design drawings submitted for construction permits and geometric data from aerial photo interpretation.

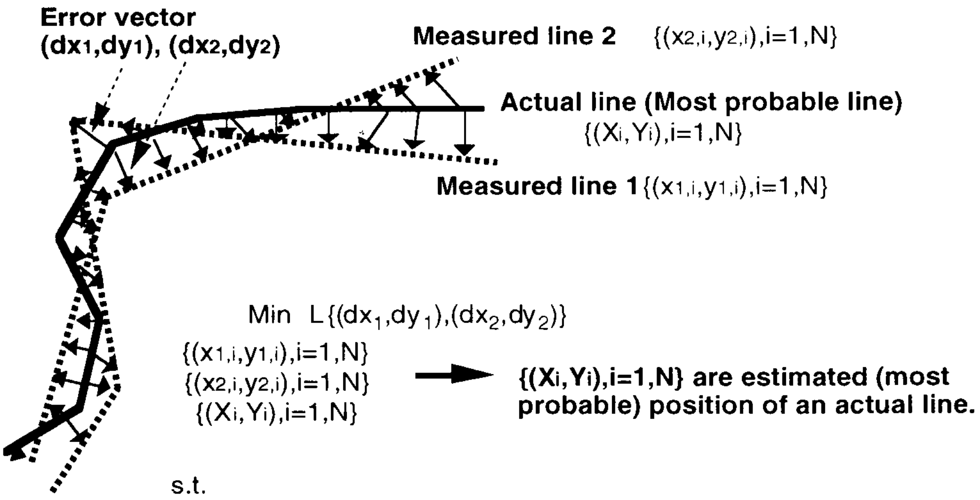
2) In estimating most probable position of geometric features, it is necessary to fulfill constraint conditions such as "a most probable line should never cross with itself", "most probable shape and location of a house should never overlap with those of others nor cross block boundaries".

Estimation of most probable position of a point can be easily made with conventional methods of surveying, while there have been no sufficient studies for line features. **Fig.4-3** illustrates a method to estimate most probable position of a line feature. Geometric constraint conditions such as a line feature never cross with itself can be implemented by knowledge of computational geometry (due to limited space, the detail is not discussed in this paper). Since some of geometric constraint conditions are described by "or" combination, estimation of most probable position of a line feature may be very computationally heavy. To improve the efficiency, irrelevant conditions are to be excluded by spatial operations of geometric data such as checking whether "possible range of existence of a line feature (see Fig.3-6.) overlap with the others or not.

Another important factor governing computational efficiency is the accuracy of initial position of estimated lines. **Fig.4-4** shows a method to set initial position of line features. Even in areas where repeated measurement and no measurement (only old data) are mixed, raster operation such as thinning can provide initial position of line features rather accurately and efficiently. In thinning process, by controlling "thinning speed" based on variance of measured lines, accuracy can be improved. Buffer areas with larger variance of positional errors should be thinned more rapidly.

5.CONCLUSIONS AND FURTHER STUDIES

1) A model to represent geo-objects with spatio-temporal uncertainties is proposed. The model can serve as a basis of developing GIS databases, because basic spatial operations are also



$$\min_{\substack{\{(x_{1,i},y_{1,i}),i=1,N\}\\ \{(x_{2,i},y_{2,i}),i=1,N\}\\ \{(X_i,Y_i),i=1,N\}}} L\{(dx_1,dy_1),(dx_2,dy_2)\} \longrightarrow$$

$\{(X_i,Y_i),i=1,N\}$ are estimated (most probable) position of an actual line.

s.t.

- $\{(x_{1,i},y_{1,i}),i=1,N\}$ and $\{(x_{2,i},y_{2,i}),i=1,N\}$ should be on each measured lines respectively.
- Distances between (X_i,Y_i) and (X_{i+1},Y_{i+1}) for $i=1,N-1$ are all identical.
- The order of $(x_{1,i},y_{1,i})$, $(x_{2,i},y_{2,i})$ and $(X_i,Y_i),i=1,N$ along each lines should be maintained
- Every node of line 1 and 2 should be on $\exists$ i. $(x_{1,i},y_{1,i})$ and $\exists$ j, $(x_{2,i},y_{2,i})$.
- If estimated line should be a part of an existing line, start and/or end points should be on the existing line.
- A most probable line should never cross with itself.

Additional constraint conditions (if necessary)

- A most probable line should never cross a boundary of a given polygon
- A most probable line should pass on a given point etc.

$$L\{(dx_1,dy_1),(dx_2,dy_2)\} = \vec{e}_1^{\,t} \cdot A_1 \cdot \vec{e}_1 + \vec{e}_2^{\,t} \cdot A_2 \cdot \vec{e}_2$$

$$\vec{e}_1 = \begin{pmatrix} dx_{1,1} \\ dy_{1,1} \\ \\ dy_{1,N} \end{pmatrix} \quad \vec{e}_2 = \begin{pmatrix} dx_{2,1} \\ dy_{2,1} \\ \\ dy_{2,N} \end{pmatrix}$$: Error vectors for line 1 and line 2

$$A_1 = \begin{pmatrix} \sigma_{x_{1,1},x_{1,1}} & \cdots & \sigma_{x_{1,1},y_{1,N}} \\ \vdots & \ddots & \vdots \\ \text{sym.} & & \sigma_{y_{1,N},y_{1,N}} \end{pmatrix} \quad A_2 = \begin{pmatrix} \sigma_{x_{2,1},x_{2,1}} & \cdots & \sigma_{x_{2,1},y_{2,N}} \\ \vdots & \ddots & \vdots \\ \text{sym.} & & \sigma_{y_{2,N},y_{2,N}} \end{pmatrix}$$

: Variance/covariance matrix of error vectors 1 and 2

Fig.4-3 Estimation of Line Position using Maximum Likelihood Method

provided to handle geo-objects represented by the model.

2) A method is proposed to estimate status of geo-objects which only emerge and are removed using observational data from multi-sources. However it is necessary to test the proposed method using actual data on changes of housed and other urban objects. And further extension

of the method is required to estimate status of more dynamic geo-objects.

3) A method to measure positional discrepancies of line features is proposed. By applying the method, most probable position of a line feature can be estimated taking several geometric constraint condition into account. With the method, repeated or overlapped line feature data can be merged based on statistical background. However, an assumption that error vector have multi-dimensional normal distribution have to be validated.

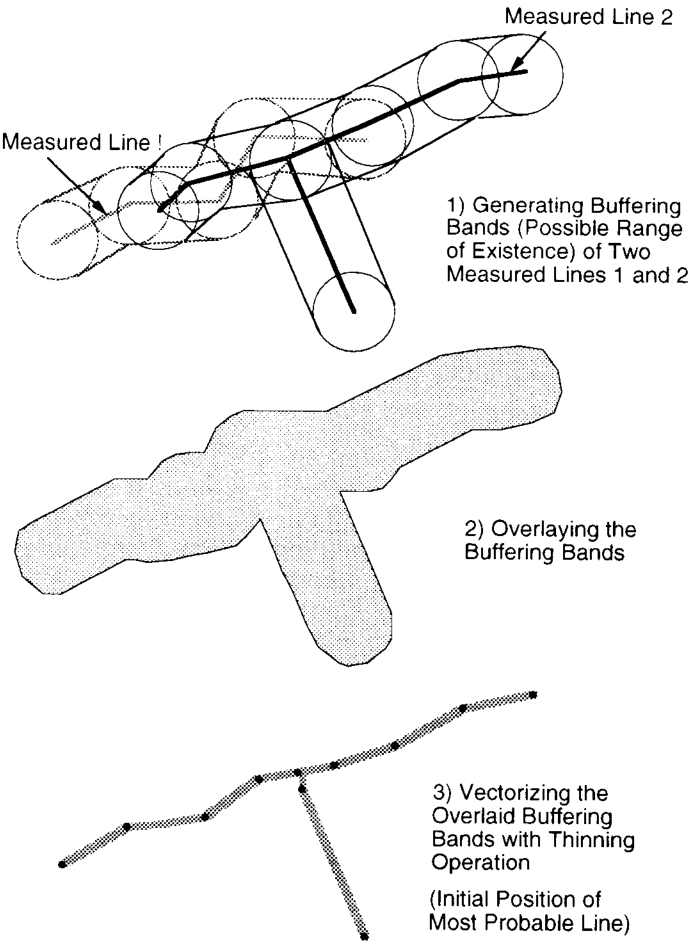


Fig. 4-4 Setting Initial Position of a Most Probable Line using Rater Operations

6.REFERENCES

Console, L. and Torasso,P. (1993): Temporal constraint satisfaction on causal models, Information Sciences, vol.68, pp.1-32.

Dechter,R. et al.(1991): Temporal contraint networks, Artificial Intelligence, vol.49. pp.61-95.

Langran, G.(1992):Time in Geographic Information Systems, Taylor and Francis

Goodchild,M.F.,and Dubuc,O.,(1987);A model of error for choropleth maps,with applications to geographic information systems, *Auto-Carto 8*, p.165.

Molenaar,M.(1989): Single Valued Vector Maps ---- A Concept in GIS, Geo-Informations-Systeme, vol.2,no.1, pp.18-26.

Shibasaki,R.(1993): A framework for handling geometric data with positional uncertainties in a GIS environment, *GIS: Technology and Applications*, Hongjun Lu and Beng Chin Ooi eds. World Scientific, pp.21-35. 1993.6

Kadowaki,K.and Shibasaki,R. (1992): Measurement and Evaluation of Positional Accuracy of Line Features in Digital Map Databases, *Proc. of Spring Conference of JSPRS*, pp.71-73. (in Japanese)

Closing Session

ADVANCED TECHNIQUES FOR THE STORAGE AND USE OF DIGITAL GEOGRAPHIC DATA: WHERE HAVE WE BEEN AND WHERE ARE WE GOING?

D.J. PEUQUET
Department of Geography, 302 Walker Bldg.
The Pennsylvania State University
University Park, PA 16802
Email: peuquet@geog.psu.edu

ABSTRACT

This paper reviews, in general terms, the development of GIS technology since its beginnings in the 1960's, particularly with respect to changes in the general approach toward research in the area of spatial data modelling. Some observations are made regarding how far we have come and where we are going. Highlights of the workshop are discussed within the context.

1. WHERE HAVE WE BEEN AND WHERE ARE WE NOW?

Beginning in the 1960's, the first GISs were developed predominantly by or for governmental agencies in order to address some geographical data handling task that was too large or complex to be handled by traditional, manual methods. Perhaps the best-known early GIS was the Canada Geographic Information System (CGIS) developed by Environment Canada for compiling and maintaining a national natural resource inventory (Tomlinson, 1973). Although early GISs contained many innovations through sheer necessity, the most expeditious approach for implementing automated geographic data storage and analysis capabilities was to simply translate traditional methodologies into the computer context whenever possible. Thus, line-for-line translations of the paper map in what became known as the vector format became the most common form of data representation. The other form of geographic data representation, raster or gridded format, was driven by input and output hardware: The first digital maps were produced on the line-printer using character positions as positions on an xy grid. The advent of raster graphics CRT's and the availability for the first time of large quantities of data captured directly as raster-format digital images both served to greatly reinforce the attractiveness of the raster format.

The well-known raster-vs-vector debate of the late 1970's and the 1980's was a direct result of the different functional advantages and disadvantages offered by these two data representational approaches. Given the original rationales behind the respective original popularity of each approach, the entire raster-vs-vector debate can be viewed in retrospect as an attempt to choose between: (1) the conceptual comfort of a traditional spatial representation (i.e. the paper map) and associated methodologies, and (2) yielding to the representation seemingly more compatible with digital technology. Why did the raster-vs-vector debate continue for so long, and indeed, why did this become such a contentious issue in the first place? The answer seems to be primarily the push toward GIS technology as an immediate solution to spatial data handling and analysis needs.

1.1 Functional Needs vs. Capabilities: A Widening Gap

In the 1990's, there are now a series of global-scale problems that are being viewed with increasing urgency: The tropical rainforests are rapidly being cut down, yet the demands of rapidly growing populations in these regions are difficult to meet; the end of the Cold War and the resultant emergence of new democracies has made international relations and the world economy more unpredictable. Such problems involve complex natural and human systems that are also often intertwined. These complex systems are certainly very difficult to understand and predict. Because of the spatial data handling and analysis potential of GIS, their use is critical as an enabling technology for addressing these problems. GIS technology has become widely adopted as a practical tool for dealing with a wide range of spatial analysis tasks. Use of GIS technology is commonplace for addressing local and regional-scale, as well as global scale applications. As part of this general trend, a number of federal governments are moving rapidly forward in the development of on-line, nationally-integrated databases. One example is the current inter-agency effort within the U.S. to incorporate a wide variety of remote sensed imagery, cartographic and ancillary data within an integrated, global database as part of an emphasis on studying global change (Committee on Earth and Environmental Sciences, 1992; Committee on Earth and Environmental Sciences, 1994).

The continued dominance of a technically oriented, immediate problem-driven mode in the GIS research community has fostered reliance on a fragmented gathering of approaches inherited from cartography, imposed by hardware, or borrowed from other computer-related fields. As a result, there is now a lack of any coherent conceptual framework or cohesive body of GIS theory. This, in turn has meant that advances in the overall functional capabilities of GIS have been slow. Increased functionality has also characteristically been accompanied by increased conceptual complexity, making GIS progressively more non-intuitive for the user. We are now faced with a situation where: (1) the initial promise of true analytical capability within an integrated data handling environment has still not been realized almost thirty years after the first GIS became operational, and (2) handling the large, shared multiscale, heterogenous databases now required for addressing urgent social, economic and environmental problems is far beyond current capabilities (Anselin & Getis, 1992; Goodchild, 1992; Peuquet, 1988). In response, the current overall approach within the GIS research community is finally shifting to a more comprehensive view with a more theoretical orientation - in other words, finding *solutions*.

Extending the capabilities of GIS in order to deal with large, multi-faceted problems implies the need for multiple representations within an accepted overall theoretical framework. The specific conceptual representation used for performing a given spatial task varies for different types of tasks, even if the type of information employed is the same. For example, when asked to describe the overall physical layout of the town or city in which one lives, streets are conceptually viewed in an overall pattern or grouping of patterns. On the other hand, when asked for directions for navigating from one place to another, we tend to think in terms of streets as linear routes with either landmarks or individual linkages. Representations may also differ considerably from one individual to another (some people navigate via landmarks, others by route linkages), yet we manage to communicate spatial information.

From an analysis and modelling perspective, the manner in which geographic information is presented and used for scientific investigations is similarly linked with the problem at-hand. For example, the spatial information in routing problems is typically represented as links between places denoted as points. In market area studies, a continuous surface is divided into thiessen polygons around point locations. For other purposes, these same places may be represented as polygonal objects that are locationally defined by explicit boundaries. The inherent structure within the information, such as interrelationships among various elements, must also be preserved as the specific intent of analysis is to derive information about this inherent structure. Nevertheless, the suitability of any representational scheme for use with a given analysis technique is often determined intuitively. Since the results of any analysis can be greatly influenced by how the phenomena under study are viewed, the development of a more integrative approach with more predictable results is essential.

2. WHERE ARE WE GOING AND HOW DO WE GET THERE FROM HERE?

2.1 Beyond the Map

A significant number of papers at this workshop are representative of a more integrative and longer-term approach that is beginning to take precedence within the GIS research community. For example, Gahegan describes a task-oriented context for representation of geographic features within satellite images (Gahegan, 1994). The stated goal of this research is to automatically provide the appropriate data representation. He describes three types of representations: image (raw), thematic (classified) and feature-based. Molenaar describes a syntactic approach for dealing with inexactness in geographic features and their relationships (Molenaar, 1994). He proposes a multi-representational scheme that incorporates both thematic and feature-based views. In his scheme, features can be represented in either vector or raster form, the choice, again, being task-dependent. Gaede and Rickert describe a hierarchical feature-based representation together with a query processing scheme that has feature-based and geometric-based aspects (Gaede, 1994). Other papers presented at this workshop also reveal this integrative and task-oriented approach in dealing with other aspects of GIS.

The realization of systems that are capable of dealing with multi-scale, multi-faceted problems will require a technological leap, which, in turn, will require a widespread rediscovery of the representation of space as a fundamental theoretical issue within geography and related disciplines as well as in the GIS community. This entails "seeing beyond" the paper map, as discussed by Chris Gold, and examining new spatial metaphors (Gold, 1994). Nevertheless, we certainly should not abandon the cartographic metaphor, as discussed by Hirtle (Hirtle, 1994), but rather re-examine the fundamental characteristics of spatial representation from a broader, more integrative and interdisciplinary perspective.

2.2 An Integrated Approach to Data Representation

The most universal and well-known representation scheme for geographic phenomena is the map. The map can be viewed both as a graphic *image* and as a geometric *structure* in graphic form. An image is a natural visual object characterized by the variation of lightness and darkness, pattern, and possibly variation in color. It may or may not convey meaning, as in the case of an abstract painting. A geometric structure, on the other hand, is an

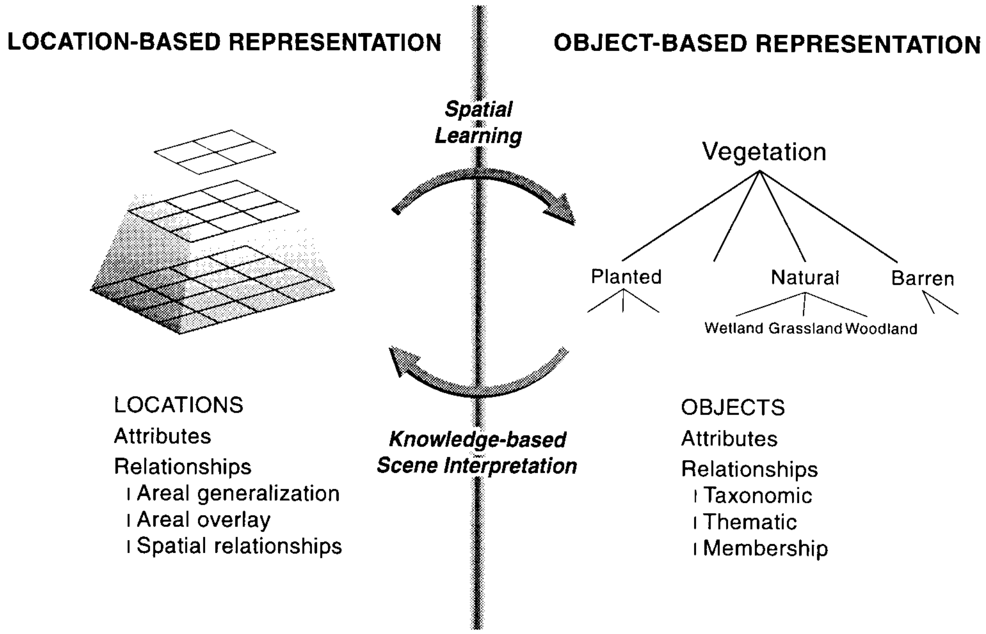


Figure 1: The Dual representational framework (from Peuquet 1988)

properties are stored and relationships act as operators on these entities and properties. Considering the feature-based form of representation, the basic logical entity is an "entity" in the every day sense, i.e., a lake, street, or street segment. Properties of features are things that describe or characterize those features. Examples of properties that could be used to describe features would include size, shape, orientation, color and height. Location may also be a property associated with a feature, such as latitude, longitude, or street address or both. In the location-based form of representation, it is noted that *locations* are the basic logical "entities" or units of reference. Thus for each location in the image, we can assign properties, and locations have relationships to other locations. In this view locations also have properties or attributes associated with them that would include brightness, darkness and color. In the broader context of general spatial representations, they may also include such things as temperature and height. These represent properties that are directly observable in the seen world and may or may not be the properties that could also be used to describe objects. Relationships in a location-based context take on a very special character; these are spatial relationships, such as "contains" or "east of."

A key component of such an integrated scheme is to develop an understanding of how the respective representations interact with each other in a functional sense. This is necessary in order to overcome the fractured approach of the past. It is particularly vital in this dual framework if we are not to find ourselves with little more than a reconstruction of the old raster-vector dichotomy.

Moving toward this necessary next-step, much insight can be gained from Marr's notion of the "primal sketch" (Marr, 1982). According to Marr, primitive features, as regions or lines, are extracted from the raw (uninterpreted) image or a series of raw images to obtain (using a graphic metaphor literally) a sketch-like representation of the image that Marr called the "primal sketch." This primal sketch consists of image elements (lines, curves, areas, etc.) that are labeled with attribute or property values. It is the first step in delivering meaning and cognitive *structure* from the *image*. Grouping processes may then be used to assemble more complex elements. These elements eventually include individual entities that can be derived reliably from the image. These are stored as "tokens" that can be assigned specific values for attributes, such as size or orientation.

The overall mechanism proposed by Marr is that descriptive primitives are built up, from the most detailed level through successive groupings, producing hierarchies of entities and spatial patterns. This is an abstraction process where the tokens refer to increasingly abstract properties of the image at higher levels of the hierarchy. How to determine "meaningful" groupings can almost never be determined directly from the scene (i.e., the observed data). Some pre-existing knowledge concerning the nature of the given phenomena involved must be employed. The pre-existing knowledge or conceptual view is also organized and used differently from the "raw Image" or seen view (Marr, 1982).

Although Marr's overall scheme must be recognized as a simplified, high-level description of a complex process, it essentially ignores one key aspect of concern here, viz., he describes a one-way process in which we start with the imaging and successively assemble features and then associate them with stored concepts. More realistically, this process works both ways. Knowledge about what is expected in the scene (from stored concepts) is also used to speed interpretation of the raw image; it can function both in allowing inferential "leaps" in feature identification and in selecting and ordering elements in the scene for subsequent interpretation.

2.3 Relationships as Operators

Given the emergence of some overall consensus on interdependent feature-based and location based representations, a general categorization of relationships follows from this. As is true of Relational Database Management Systems, relationships are also treated as *operators* on stored data concerning features or locations. At this workshop, these were referred to as feature based and geometric-based by Gaede and Riekert (1994). The query classification scheme offered by Schenkelaars (1994) suggested thematic (feature-based), geometric (location-based), as well as topological, as a separate class of queries that combine the feature-based and location-based views.

In a dual representation, it is also possible to classify all potential questions into two basic types:

(1) Given a specific feature or set of features, where is it located? For example: Where is Delft?

(2) What object or objects are present at a given location? For example: Is there a city located at coordinates 52°N, 4° 22'E?

The first question would be answered by utilizing the feature-based portion of the representation and the second question by utilizing the location based portion of the representation. These questions correspond to the object-based and location- or image-based views respectively.

If for the sake of discussion we continue to assume a distinctly dual scheme with a clear distinction between the two types of views, then each of the two types of views have three types of relational operators. These two sets can be seen to correspond to a particular axis in the diagram over which they operate. Taxonomic operators include generalization/aggregation and allow association among entities up and down a feature hierarchy. Areal generalization operates in a similar manner over varying geographic scales. Membership relationships in the feature-based representation and spatial relationships in the location-based representation relate various elements *across* a given common level or aggregation. Membership is, in general terms, thematic classification. Spatial relationships in this context would fall into two categories: overlay, corresponding to the mathematical boolean operation, and topological operators.

Regardless of the form of query or the specific operator or combination of operators used, there is one fundamental characteristic of geographic data that must always be taken into account - the definitions of features, locations, and their properties can be inexact, incomplete, and subject to interpretation depending on the application context. This is also true of relationships *between* features or locations. As noted by Gapp at this workshop, insights into a solution for dealing with this can be gained by investigating previous work in the fields of perceptual psychology and linguistics in order to integrate visual and verbal conceptual space with more commonly used mathematical geometrical conventions (Gapp, 1994). As a notable example of such work, Jackendoff (Jackendoff, 1983) examined the question of how meanings can be attributed to words. For our purpose, we assume that words for entities, properties and relations are symbolic tokens for the things themselves and therefore the process of attaching meanings to words is essentially the same as attaching meanings to entities. To summarize Jackendoff's discussion, at least three sorts of conditions are needed to adequately describe word meanings. First are the *necessary* conditions. For example, "lake" must contain the necessary condition of also being a hydrologic feature. Secondly, there are graded conditions. These specify a central or threshold value for a continuously variable property, such as elevation for "mountain." These are called *centrality* conditions. Third are *typicality* conditions, such as we would normally associate fresh water with "lake" (although some contain salt water). These conditions can also be extended - describe other elements, including spatial relationships. Inexact relationships of other types, such as the taxonomic relation 'is-a', could also be described in terms of such conditions.

Although this offers some ability in the definition or description of relationships to deal with variations and gradations in meaning, more is needed for evaluating and validity of a relationship in a given case. Some functional criteria were enumerated by McClelland for assigning meaning to words (McClelland, Rumelhart, & Hinton, 1986). These can be interpreted within the context of spatial relationships as follows:

1) *Graceful degradation* - This is effectively 'benefit of the doubt'. Given that there is often no entity that satisfies the relational qualifier in the strictest sense, all entities that marginally qualify are accepted. Thus, in the question regarding a spatial relationship, 'Is there a city in central Pennsylvania?' we may answer in the affirmative, saying that State College (definitely in the center of the state) qualifies because of its location even though whether it would quality as a "city" given its lack of size compared to Philadelphia in the extreme eastern side of the state. Graceful degradation becomes particularly important when there is a combination of several qualifiers that must be simultaneously satisfied and must rely on a combination of centrality and typicality conditions.
2) *Default assignment* - This is the assertion that we can also assume some properties and values of properties for partially known features on the basis of similar or "higher-level" features. This characteristic is also important with respect to the location-based view. In this context, properties of an unknown location or group of contiguous locations (i.e., an area) can be inferred or interpolated from the value of that attribute at adjacent or containing locations. This is based on the assumption that spatial surfaces are continuous. For example, in the question, 'Is Delft

mountainous?' we could certainly answer "No" simply on the basis of the knowledge that Delft is in Holland, which definitely does not contain any mountains.

As with the necessary, centrality and typicality definitional rules, these rules for application would also apply to other types of inexact relationships. Together, such rules form a conceptually simple yet potentially powerful means of handling inexactness in the conceptual formalization of spatial relationships. This is very similar to the concept of Inheritance within the field of artificial intelligence, except that Default Assignment can be applied in the case of any "similar" features, not just within parent/child feature types.

This brief sampling shows that there are mechanisms already developed in a variety of fields that offer significant promise for use in extending the traditional geometric definitions of individual spatial relationships so that their meaning and evaluation is more 'natural' in human terms.

3. CONCLUDING REMARKS

The representational scheme that is emerging by integrating principles and theories from a number of disciplines provides insight for the first time on the *functional* relationship between location-based and feature-based views that have existed implicitly in GIS for many years. The high degree of parallelism shown between the location-based and feature-based views and the limited number of types of relationships seem to indicate that a unified framework for representing geographic phenomena need not be as complex as had been previously anticipated. The more theoretically-oriented, longer-range approach that has developed within the GIS research community also seems to be offering hope of, if not closing the gap, then at least of keeping up with demands in GIS functional capabilities for solving real-world problems. Nevertheless, it remains a multifaceted theoretical issue. Toward this goal, there is certainly need for research to continue in these basic areas. Perhaps the most important of these areas is the definition of the relational operators.

Spatial relationships are unique to locational information. Spatial relationships and how they interact need to be refined so that formal mechanisms for combining operations can be developed. Another critical area of research is to examine the functional linkages between the locational and feature representations. This is fundamental to deriving formalized transformation procedures between varying representational forms. The properties attributable to entities and relationships in general must also be better defined and understood before a spatial algebra for combining relationships can be developed that is useful in either a manual or automated context. Such an algebra would be an extremely powerful tool for describing spatial pattern and process, particularly when it is based on a theoretical representational framework.

Another major element in the representation of geographic space that has not been addressed until now is temporal dynamics. Current GIS, and the data models they employ, exist only in the "present." Information can be added-to and modified over time, but a sense of past history is not maintained. This, again, may be attributed to the traditional (i.e., static) cartographic paradigm for spatial data representation in GIS.

There has been broad-based interest on the practical issue of functional databases that represent geographic time-series data only in the past few years. The fundamental characteristics of relations with temporal dynamics therefore represents another major and needed area of research. At a high level of abstraction, the extension of current representations to include time would seem at first glance to be a trivial task, but recent research on this topic shows that this is not the case (Hazelton, 1991; Kelmelis, 1991; Langran, 1992; Peuquet, 1994). The additional dimension of time adds a corresponding additional dimension to the representation. Besides the corresponding addition of purely temporal relationships, such as "during", there are also relations which involve both temporal and spatial aspects, such as "through" and "around". These relations involve a variation of both location and time as it pertains to a given feature. Adding a dimension to the representation for time seems therefore to imply that relations can function as mappings on two axes instead of just one.

In summary, the switch in focus within the GIS research community away from immediate problem-solving and toward longer-term, theoretical issues is already beginning to show results, although there is still a long way left to go. Nevertheless, this switch in focus can be seen as indicative of the maturation of GIS as a science. This maturation process has already required a fundamental change in how we view information relating to geographic space.

REFERENCES

Abdelmoty, A., & Williams, M. H. (1994) *Approaches to the Representation of Qualitative Spatial Relationships for Geographic Databases*, Paper presented at the Advanced Geographic Data Modelling, Delft, The Netherlands

Anselin, L., & Getis, A. (1992) Spatial Statistical Analysis and Geographic Information Systems. *Annals of Regional Science*, *26*, 19-33

Blades, M., & Spencer, C. (1986) The Implications of Psychological Theory and Methodology for Cognitive Cartography. *Cartographica*, *23*(4), 1-13

Committee on Earth and Environmental Sciences (1992) *The U.S. Global Change Data and Information Management Program Plan* . Office of Science and Technology Policy

Committee on Earth and Environmental Sciences (1994) *Our Changing Planet: The FY 1994 U.S. Global Change Research Program* . Office of Science and Technology Policy

Freeman, H. (1974) Computer Processing of Line-Drawing Images. *Computing Surveys*, *6*, 57-97

Gaede, V., & Riekert, W.-F. (1994) *Spatial Access Methods and Query Processing in the Object-Oriented GIS GODOT*, Paper presented at the Advanced Geographic Data Modelling, Delft, The Netherlands

Gahegan, M. (1994) *Formal Semantics for the Integration of Image Data into Geographic Information Systems*, Paper presented at the Advanced Geographic Data Modelling, Delft, The Netherlands

Gapp, K.-P. (1994) *A Computational Model of the Basic Meanings of Graded Composite Spatial Relations in 3D Space*, Paper presented at the Advanced Geographic Data Modelling, Delft, The Netherlands

Gold, C. M. (1994) *The Interactive Map*, Paper presented at the Advanced Geographic Data Modelling, Delft, The Netherlands

Goodchild, M. F. (1992) Geographical Information Science. *International Journal of Geographical Information Systems*, *6*(1), 31-45

Hazelton, N. W. J. (1991) *Integrating Time, Dynamic Modelling and Geographical Information Systems: Development of Four-Dimensional GIS*, Unpublished unpublished Ph.D. dissertation, Dept. of Surveying and Land Information, The University of Melbourne

Hirtle, S. C. (1994) *Towards a Cognitive GIS*, Paper presented at the Advanced Geographic Data Modelling, Delft, The Netherlands

Jackendoff, R. (1983) *Semantics and Cognition*. Cambridge, MA: The MIT Press

Kelmelis, J. (1991) *Time and Space in Geographic Information: Toward a Four-Dimensional Spatio-Temporal Data Model*, Unpublished unpublished Ph.D. dissertation, The Pennsylvania State University

Langran, G. (1992) *Time in Geographic Information Systems*. London: Taylor & Francis

Marr, D. (1982) *Vision*. New York: W.H. Freeman Co.

McClelland, J., Rumelhart, D., & Hinton, G. (1986) The Appeal of Parallel Distributed Processing. In D. Rumelhard & J. McClelland (Ed.), *Parallel Distributed Processing, Explorations in the Microstructures of Cognition*(Vol. 1, pp. 3-44). Cambridge, MA: The MIT Press

Molenaar, M. (1994) *A Systactic Approach for Handling the Semantics of Fuzzy Spatial Objects*, Paper presented at the Advanced Geographic Data Modelling, Delft, The Netherlands

Peuquet, D. J. (1988) Representations of Geographic Space: Toward a Conceptual Synthesis. *Annals of the Association of American Geographers*, *78*(3), 375-394

Peuquet, D. J. (1994) It's About Time: A Conceptual Framework for the Representation of Temporal Dynamics in Geographic Information Systems. *Annals of the Association of American Geographers*, *84*(3), 441-461

Schenkelaars, V. (1994) *Query Classification, a First Step Towards a Graphical Interaction Language*, Paper presented at the Advanced Geographic Data Modelling, Delft, The Netherlands

Tomlinson, R. F. (1973) *A Technical Description of the Canada Geographic Information System* (unpublished report)