Extended Abstract

REPRESENTATION OF GEOMETRIC OBJECTS AS SET OF INEQUALITIES[1]

Andrew Frank, Peter Haunold, Werner Kuhn[2], Gabriel Kuper[3]

1. Introduction

Most GIS but also CAD systems represent the objects by their boundaries. Wireframes, winged edges and similar methods are widely used. With these methods, object geometry is defined by a collection of boundaries (surfaces or edges); the relations between these boundaries are usually explicitly stored [Frank, 1987; Oliver, 1988].

As an alternative, objects can be represented by inequalities which describe half-planes (half-spaces in 3D). There are several papers [Rigaux, & Scholl, 1995] see there for earlier papers] which propagate this idea and explore the expressive power and the demands the resolution of such systems of constraints poses. Systems of Linear Constraints (Linear Constraints [Frank, & Wallace, 1995; Freeston, Kuper, & Wallace, 1995]) are relatively easy to solve and powerful methods are known. These methods have been used for a long time in Operation Research and highly efficient programs for their practical solution exist.

If the efficient methods to solve linear constraints can be applied to the geometric problem, much can be gained practically and theoretically. Many of the algorithms for constraint resolution are simpler than computational geometry algorithm and often their structure is also easier to understand and program. For example, the constraints describing an area do not depend on a particular order, but can be reordered to improve processing; the intersection of two areas is found by simply merging the two lists of constraints.

This paper explores whether this alternative is practically viable for the use in GIS. The application of this theoretically attractive representation method depends on practical conditions:

- how much storage is used,
- can large data collections be processed effectively, and
- are the most important operations of GIS implemented efficiently.

This paper takes GIS as an example and uses their typical property [Frank, & Kuhn, 1995] as a benchmark to compare a representation of geometry by inequalities.

2. Storage

Considering the description of a region as a list of inequalities seems to require large amounts of storage, but this is not correct, as will be shown here. In most GIS areal data in form of partitions of space dominate the storage requirements and these are discussed here primarily.

An inequality is represented as two signed real numbers, which is the same as in a regular vector representation (where n points represent the n edges forming a polygon, i.e. for each line in the polygon one coordinate pair is stored). For a closed polygon, there is the same number of inequalities as there are points in the boundary of the area. The data structure to store inequalities can be as simple (or complex) as for the vector representation (e.g. a linked list). For unconnected areas there is no difference in the amount of storage required between a vector representation as closed polygon or a list of inequalities.

$$ax - by \le c \qquad (1)$$

$$x - \frac{by}{a} \le \frac{c}{a} \qquad (2)$$

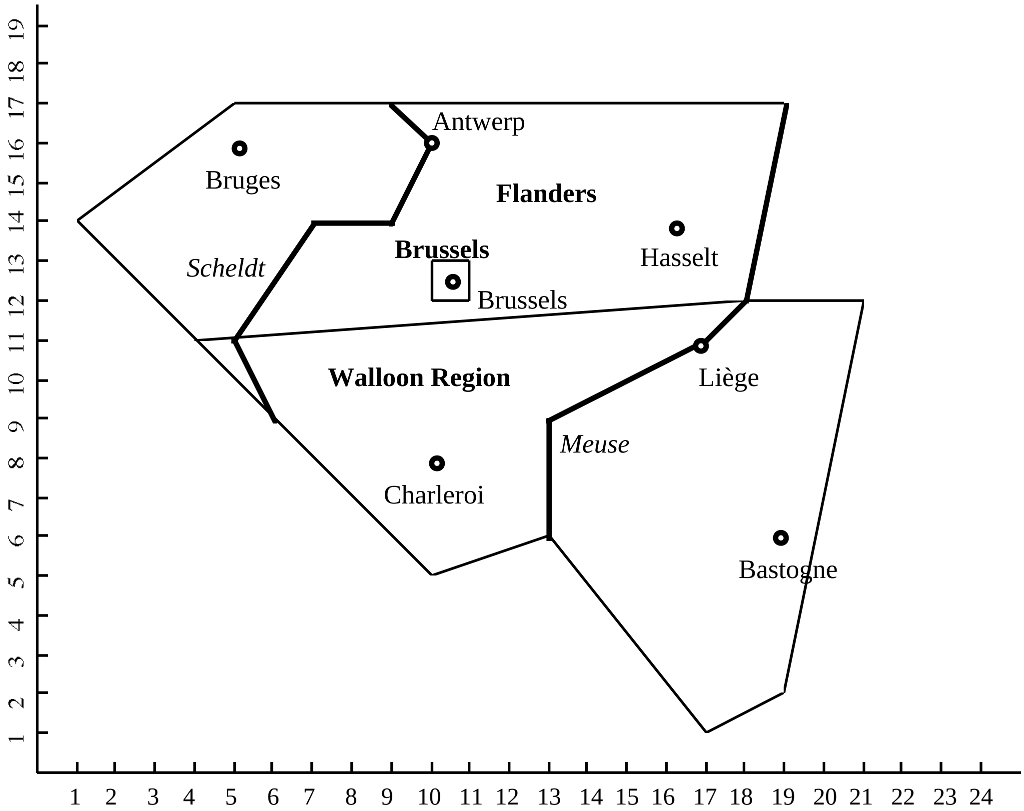
To control point precision better the form (1) may actually be easier than (2).

---

[1] This work is part of the cooperation within the Esprit CONTESSA project.

[2] Address of Andrew Frank, Peter Haunold and Werner Kuhn:
Dept. of Geoinformation, Technical University Vienna, Gusshausstrasse 27-29, A-1040 Vienna,
email: {frank, haunold, kuhn@geoinfo.tuwien.ac.at}

[3] Address of Gabriel Kuper:
European Computer Research Center, Munich, Arabellastr. 17, D-81925 Munich
email: gabriel.kuper@ecrc.de



Representation using inequalities:

Regions

Name	*Geometry*
Brussels	$(y \leq 13) \wedge (x \leq 11) \wedge (y \geq 12) \wedge (x \geq 10)$
Flandres	$(y \leq 17) \wedge (3x - 4y \geq -53) \wedge (x - 14y \leq -150) \wedge (x + y \geq 45) \wedge$ $(4x - y \leq 78) \wedge (\neg((y \leq 13) \wedge (x \leq 11) \wedge (y \geq 12) \wedge (x \geq 10)))$
Walloon Region	...

Cities

Name	*Geometry*
Antwerp	$(x = 10) \wedge (y = 16)$
Bastogne	$(x = 19) \wedge (y = 6)$
Bruges	$(x = 10.5) \wedge (y = 12.5)$
...	...

Rivers

Name	*Geometry*
Meuse	$((y \leq 17) \wedge (5x - y \leq 78) \wedge (y \geq 12)) \vee$ $((y \leq 12) \wedge (x - y = 6) \wedge (y \geq 11)) \vee$ $((y \leq 11) \wedge (x - 2y = -5) \wedge (y \geq 9)) \vee$ $((y \leq 9) \wedge (x = 13) \wedge ((y \geq 6))$
Scheldt	...

Fig. 1. Example of a spatial database [Vandeurzen, Gyssens, & Van Gucht, 1995]

Representation using boundary points - relational model:

Region

Edge ID	*From Point*	*To Point*	*Left Area*	*Right Area*
a	1	2	Belgium	Flanders
b	2	3	Belgium	Flanders
c	3	4	Walloon Region	Flanders
d	4	5	Belgium	Flanders
e	5	1	Belgium	Flanders
...				

Tab. 1. Relational representation for one region only

For actual GIS applications, where the distribution of land to owners is recorded (so called parcel data), categorical coverage [Chrisman, 1987] or partitions of space (figure 1) and regions must share boundaries. For each line (or inequality) a pointer to the left and right area must be included (table 1). Edges are represented by pointers to the start and end-point. For nodes with more than two adjacent edges, the storage for the point coordinates are shared with more edges, reducing average storage requirements (compensated by a need for pointers).

Practical example: An actual parcel map at the 1:2000 scale from the Austrian cadastre shows in an area of 15 by 15 cm:

48 Parcels	1.6 Edges/Node
283 Nodes	5.9 Nodes/Parcel
180 Edges	3.8 Edges/Parcel

Formula (1) gives a table structure for the edges as in figure 1. This results in a storage requirement per edge of 2 pointers and 3 integers (corresponding to one coordinate value). This compares with the 4 pointers to the table of coordinates and the 2 coordinate values for each point (of which come on average 1.6 per edge). If pointers and coordinate values take the same amount of space, this gives 5 values per edge for inequalities and 5.6 values for the vector representation. This is for practical purposes the same storage amount.

Id	**a**	**b**	**c**	**left**	**right**
a	0	1	17	Belgium	Flanders
b	3	-4	-53	Belgium	Flanders

The most obvious difference is the lack of order information in the Linear Constraints and of explicit representation of points.

- Because Linear Constraints does not require order, an overlay operation to find the common areas for two tessalations, is only merging the two sets of Linear Constraints. The computation of an overlay does not require any floating point operations; these become only necessary at the end to calculate the boundary points (if this is required). Points for intermediate results are not calculated and the inequalities for the boundaries are not changed during calculation (this excludes the creeping of points during an overlay operation [Guevara, 1985]).
- Boundary points (corners) are very important for the human perception and cognition of parcels. They can be calculated from the Linear Constraints representation. Corners are typically necessary for the calculation of area using Gauss' formula, but there may exist more direct way to calculate area from Linear Constraints.

We conclude this section with the observation that the representation of a data set for a typical GIS application, e.g. representing cadastral parcel geometry, in Linear Constraints requires a comparable amount of storage as would a pointer-based vector representation.

3. Effective processing of large data collections

The data collection for a GIS is large. In [Frank, & Kuhn, 1995; Goodchild, 1990] a typical value of 10 - 400 Gigabytes is expected. It is thus completely excluded that for deciding a simple ‘point in polygon’ or a ‘range’ query the complete data collection can be checked. It is necessary to build a spatial access method [Ester, Kriegel, & Xu, 1995; Frank, & Kuhn, 1995; Hjaltason, & Samet, 1995].

3.1. Requirements for Spatial Access Methods

Spatial access methods allow to select quickly the data relevant for a problem and exclude the large majority of the stored data from consideration and thus from processing. The spatial access method abstract from detailed properties of the geometry of an object, but preserve enough information that one can decide quickly if an object (or large subsets of objects) can be excluded from consideration. The object geometry is typically represented as minimal bounding rectangles (MBR) or similar compact generalizations of the form and location of an object. Methods using other geometric forms, e.g. circles, were proposed in [Finke, & Hinrichs, 1995]. Processing is more selective but slower than for the rectangular methods. The advantage does not seem large enough to be tried in practice.

Spatial Access Methods require that for each object to be selected based on location a simple geometric abstraction can be supplied (i.e. a Minimal Bounding Rectangle). The selection operation is then based on this simpler geometric abstraction and must produce a superset of the desired result (i.e. false positives are allowed, but no false drops).

3.2. Minimal Bounding Rectangles for Linear Constraints

For inequalities minimal bounding rectangles do not apply, but the rectangles can be formed around the areas (i.e. a collection of inequalities). Object selection is then based on MBR and results in the collection of areas to consider.

The well known tree structures using the minimal bounding rectangle (MBR) representation of stored objects can be used, even if the objects are stored as a set of linear inequalities [Ester, Kriegel, & Xu, 1995; Güting, de Ridder, & Schneider, 1995; Hjaltason, & Samet, 1995]. Selection based on comparing the MBR retrieves all the inequalities necessary for the standard operations.

In principle, MBR could also be associated with the edges, both in the vector or in the Linear Constraints representation. The difference is only that the MBR cannot be calculated from the Linear Constraints alone, but the area it bounds and all the constraints defining it must be available. The MBR then represents the ‘active’ part of the constraints, namely the piece that is forming a boundary.

3.3. Generalization

Minimal bounding rectangles can be used not only for spatial access methods but generally for geometric operations. A corresponding operation is applied to the reduced representation and only objects passing this test must be included in the accurate computation. The test based on the reduced representation (e.g. MBR) is orders of magnitudes faster than the accurate test. The reduction and the two step processing is increasing performance. The accurate tests run always with very small datasets and their asymptotic behavior plays therefore a lesser role (the quick tests are nearly always linear in their time).

This generalized method of selection of relevant objects extends to the Linear Constraints representation. For most operations on objects, e.g. calculation of the intersection, the union, the boundary of a collection of areas etc., only the inequalities of the areas involved are necessary.

4. Conclusion

First considerations show that it is possible to represent the kind of data in a GIS as linear constraints. The representation as constraints affords some advantages as the representation does not depend on order as does the vector representation. Overlay processing should be faster and show less algorithmic problems. Buffer formation is much simpler and should work as well in 3D (a currently unsolved problem). Processing of inequalities generalizes to the 3D space [Raper, 1989].

Storage requirements for the Linear Constraints representation and for a vector representation have been compared for a typical case of GIS data, namely a 1:2000 cadastral plan. The storage amounts are very similar (10% more for Linear Constraints, which is an insignificant difference).

The known methods for spatial access are based on an abstract representation of geometry (typically MBR). These methods can be applied to the Linear Constraints representation as well as for a vector representation. There are no apparent reasons, why an Linear Constraints or vector representation should perform faster.

Linear Constraints is the most significant revolutionary idea for the representation of spatial data. In contrast to several proposals to slightly improve a vector representation [Finke, & Hinrichs, 1995], Linear Constraints is a completely different concept. It promises some advantages over vector processing and there are no immediately visible drawbacks, as established here.

4.1. Future Work

Much remains to be done. Detailed algorithms for the standard operations in a GIS must be developed. Operations like

- point in polygon
- determination of boundary points of a polygon
- area of a polygon
- intersection of two polygons
- intersection of two partitions (overlay)
- construction of a buffer zone around a given object

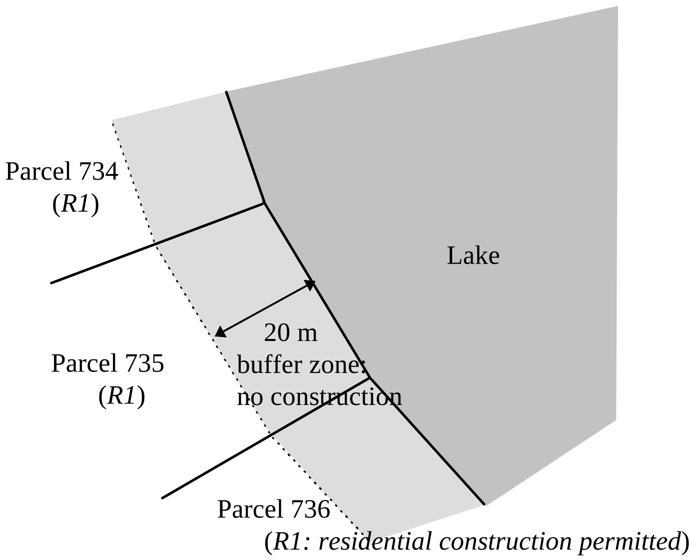


Fig. 2. Buffer zone along a lake shore

Most of these operations can be implemented efficiently using a representation with inequalities, but detailed studies considering the particular load presented by a GIS are needed. Of particular interest are buffer-zone operations (figure), which are notoriously difficult for boundary representations, but very easy for the representation with inequalities, even in 3D.

References:

Chrisman, N. 1987. Fundamental principles of geographic information systems. (Chrisman, N. R., Eds.). Auto-Carto 8, Baltimore, MA. ASPRS & ACSM. Eighth International Symposium on Computer-Assisted Cartography, pp: 32-41.

Ester, M., Kriegel, H.-P., & Xu, X. 1995. Knowledge Discovery in Large Spatial Databases: Focusing Techniques for Efficient Class Identification. (Egenhofer, M. J., & Herring, J., Eds.). Fourth International Symposium on Large Spatial Databases - SSD 95, Portland ME. Springer Verlag, Heidelberg. Lecture Notes in Computer Science, Vol. 951, pp: 67-82.

Finke, U., & Hinrichs, K. H. 1995. The Quad View Data Structure - A Representation for Planar Subdivisions. (Egenhofer, M. J., & Herring, J., Eds.). Fourth International Symposium on Large Spatial Databases - SSD 95, Portland ME. Springer Verlag, Heidelberg. Lecture Notes in Computer Science, Vol. 951, pp: 29-46.

Frank, A. U. 1987. Overlay Processing in Spatial Information Systems. (Chrisman, N. R., Eds.). Eighth International Symposium on Computer-Assisted Cartography (AUTO-CARTO 8), Baltimore, MD, pp: 16-31.

Frank, A. U., & Kuhn, W. 1995. Specifying Open GIS with Functional Languages. (Egenhofer, M. J., & Herring, J., Eds.). Fourth International Symposium on Large Spatial Databases - SSD 95, Portland ME. Springer Verlag, Heidelberg. Lecture Notes in Computer Science, Vol. 951, pp: 184-195.

Frank, A. U., & Wallace, M. 1995. Constraint Based Modeling in A GIS: Road Design as a Case Study. Auto-Carto 12, Charlotte, North Carolina (Feb. 27 - March 1).

Freeston, M., Kuper, G., & Wallace, M. G. 1995. Constraint Databases. Conference on Information Technology and its Use in Environmental Monitoring and Protection, Holloway College.

Goodchild, M. F. 1990. Tiling a Large Geographical Database. (Buchmann, A., et al., Eds.). Symposium on the Design and Implementation of Large Spatial Databases, New York, NY. Springer-Verlag.

Guevara, J. A. 1985. Intersection Problems in Polygon Overlay. Auto-Carto 7, Washington, D.C., March 11-14, 1985. ACSM/ASPRS.

Güting, R. H., de Ridder, T., & Schneider, M. 1995. Implementation of the ROSE.

Hjaltason, G. R., & Samet, H. 1995. Ranking in Spatial Databases. (Egenhofer, M. J., & Herring, J., Eds.). Fourth International Symposium on Large Spatial Databases - SSD 95, Portland ME. Springer Verlag, Heidelberg. Lecture Notes in Computer Science, Vol. 951, pp: 83-95.

Oliver, S. G. 1988. A GIS Automation of County Tax Assessor Maps and Their Application for Siting the Superconducting Super Collider in Illinois. GIS/LIS'88, Third Annual International Conference, San Antonio, Texas. ACSM, ASPRS, AAG, URISA. Accessing the World, Vol. 1, pp: 448-458.

Raper, J. 1989. GIS: Three Dimensional Applications in Geographic Information Systems. London: Tayler and Francis.

Rigaux, P., & Scholl, M. 1995. Multi-Scale Partitions: Application to Spatial and Statistical Databases. (Egenhofer, M. J., & Herring, J., Eds.). Fourth International Symposium on Large Spatial Databases - SSD 95, Portland ME. Springer Verlag, Heidelberg. Lecture Notes in Computer Science, Vol. 951, pp: 170-183.

Vandeurzen, L., Gyssens, M., & Van Gucht, D. 1995. On the Desirability and Limitations of Linear Spatial Database Models. (Egenhofer, M. J., & Herring, J., Eds.). Fourth International Symposium on Large Spatial Databases - SSD 95, Portland ME. Springer Verlag, Heidelberg. Lecture Notes in Computer Science, Vol. 951, pp: 14-28.