NCG

Nederlandse Commissie voor Geodesie   Netherlands Geodetic Commission

NCG/GIN Seminar Geo-information and computational geometry

Monday 14 November 2005 (Utrecht University)

Abstracts

1. Computational Geometry: Its objectives and relation to GIS

11:05-11:50 Marc van Kreveld (UU)

The research area of computational geometry deals with the design and analysis of algorithms for geometric problems. The analysis of algorithms involves understanding how efficiently an algorithm solves a problem. An algorithm is considered efficient if it scales well in the input size, that is, if the time needed to solve an instance with a certain input takes a few minutes, then the time needed on an input that is ten times as large does not get out of hand completely. One of the main objectives of computational geometry is finding the most efficient algorithms for all sorts of geometric problems. These problems are either abstract, or motivated from areas like computer graphics, robot motion planning, and GIS.

We will review the main concepts and ideas in computational geometry. These include efficiency analysis, intractability, output-sensitive algorithms, and approximation algorithms. The basic problems of computational geometry all have a direct or indirect use to GIS. For instance, line segment intersection by plane sweep solves map overlay, the most important GIS analysis task, Voronoi diagrams are helpful in neighborhood analysis, medial axis transforms help for buffer computation, Delaunay triangulations are widely used for terrain modeling, and geometric data structures help with efficient spatial indexing in large spatial data sets.

However, there are several reasons why computational geometry is not as useful to GIS as it could be. A first reason is that the algorithms developed in computational geometry are often rather complex, and they require a large effort to implement. This is partially because they are provably efficient for all possible inputs, and partially because geometric degeneracies must be dealt with. A second problem with the usefulness comes from the efficiency analysis, which is based on worst-case inputs to the algorithm. The theoretical efficiency of any algorithm depends on the time needed on the data set for which it is slowest. However, such worst-case data sets may be rather artificial, and never appear in real-world applications. The third problem with the applicability of computational geometry lies in the abstraction of the original problem. Many problems that come from practice require several criteria to be met simultaneously, or criteria that should be met "at least to some extent". For example, when constructing a cartogram, maintaining recognizability as much as possible is an important issue. This leads to vague problem statements, and the general practice in computational geometry is to consider well-defined, simple-to-state problems mostly.

In the last decade, several of the problems mentioned above have been addressed. Nowadays, there are software libraries with geometric primitives and algorithms, alleviating the effort to implement. They also solve the robustness issues largely. The second issue has been addressed through the development of relatively simple algorithms are inefficient in the worst case, but provably efficient under realistic assumptions on the input. Unfortunately, the third issue is still largely unresolved. Computational problems where various, partially conflicting criteria must be respected to some extent, are typical in the more advanced GIS functions. The correct formulation then becomes the first concern. Only after a study of appropriate formalizations has been done, algorithmic solutions can be studied. At the same time, implementation and testing are needed to evaluate formalizations, and refine them.

Other typical GIS aspects that should be considered in the context of computational geometry as well computation with imprecise coordinates, computations that deal with (spatial) scale of geographical phenomena and processes, and indeterminate boundaries of features. Examples of indeterminate boundaries are boundaries of mountain ranges when there is a transition zone with the lower surrounding land, and the definition of the extent (or size) of an island when there are tidal changes. Spatial and spatio-temporal data mining also offers opportunities for computational geometry.

2. **I/O- and Cache-efficient Algorithms for Spatial Data**
11:50-12:35 Mark de Berg (TUE)

Modern computer systems have a hierarchal memory consisting of a disk, main memory, and several levels of cache. The difference between the times to access these different levels of memory is quite large. This holds in particular for main memory and disk: accessing the disk is typically about 100,000 times slower than accessing the main memory. Hence, it is important to take caching- and I/O-behaviour into account when designing and analyzing algorithms. This is especially the case in GIS applications, where the data sets are often very large and do not fit into the main memory. In this talk we discuss some of the recent results that have been obtained on I/O- and cache-efficient algorithms, where we focus on algorithms and data structures for spatial data.

3. **Quad-Edges and Euler Operators for Automatic Building Extrusion Using LiDAR Data**
13:30-14:15 Christopher Gold and Rebecca Tse (University of Glamorgan, UK)

Our long-term research objective is to integrate man-made objects with the landscape, so that topological properties, such as connectedness, may be used in applications such as flood modelling. Building information should be extracted directly from LiDAR data.

Several steps are required. Building data points should be separated from the terrain data points by identifying building boundaries, either from cadastral information or by various edge-detection techniques. Bare earth topography is obtained, either from existing information or by filtering the LiDAR data. Data points inside the building boundaries are then processed to give an average building height, and to estimate the roof shape. The buildings are then constructed by treating the terrain TIN model as a CAD type b-rep surface, and performing local modification of the Quad-Edge data structure by using Euler operators. These operators permit various extrusion operations, as well as the manual insertion of bridges and tunnels.

Two approaches have been taken to the separation of building data from terrain data. Unlike the traditional approach, where walls are represented by non-vertical triangles, our first approach is to insert cadastral boundaries into our TIN by adding sufficient extra points to ensure that the boundaries follow triangle edges. Euler operators are then used to generate a second boundary (the tops of the walls) which is then extruded upwards. A more experimental method uses a second, coarser network of Voronoi cells that sample the underlying LiDAR data. These cells are then perturbed until each one is entirely part of the (higher) building or (lower) terrain. Building blocks are then extracted.

Data points interior to the building may be used to estimate the height of a supposedly flat-roofed building, or they may be used to extract the form of the roof. This is attempted by calculating the vector normals of "roof" triangles, and calculating the eigenvalues of their direction cosines. For simple roof shapes this will give the primary roof orientation, and the pattern of normals indicates the roof form. This information may then be integrated into the surface model.

4. **Algorithms for cartograms and other geo-information visualization techniques**
14:15-15:00 Bettina Speckmann (TUE)

Cartograms are a useful and intuitive tool to visualize statistical data about a set of regions like countries, states or counties. The size of a region in a cartogram corresponds to a particular geographic variable. The most common variable is population: in a population cartogram, the sizes (measured in area) of the regions are proportional to their population. In a cartogram the sizes of the regions are not the true sizes and hence the regions generally cannot keep both their shape and their adjacencies. A good cartogram, however, preserves the recognizability in some way.

There are several types of cartograms and many algorithms to compute them. This talk will give a short overview of cartogram algorithms and focus in particular on the computation of rectangular cartograms. In a rectangular cartogram each region is represented by a rectangle. Good rectangular cartograms are hard to generate: The area specifications for each rectangle may make it impossible to realize correct adjacencies between the regions.

The algorithms for rectangular cartogram construction depend on a precise formalization of region adjacencies and build upon existing VLSI layout algorithms. An implementation and various tests show that in practice, visually pleasing rectangular cartograms with small cartographic error can be generated effectively.

The talk will conclude with a brief discussion of other algorithms for specialized maps such as schematic maps and some types of thematic maps.

5. **Constrained tetrahedral models and update algorithms for topographic data**
15:30-16:15 Friso Penniga (TUD)

Efforts to expand 2D geographic data models into full 3D representations of reality often result in tricks and workarounds to model 3D features using less dimensional representations, such as surface models. This research focuses on modelling 3D features using also equally dimensional primitives, thus constructing the model of point, lines, surfaces and volumes. The 3D primitive of choice is the tetrahedron, applied in a tetrahedronized irregular network (TEN), the 3D version of the more generally known triangulated irregular network (TIN). This TEN is a very well-defined data structure which enables complex processing analogously to solving partial differential equations: separate processing on each primitive first and afterwards joining all these partial results into a final result. All topographic features will be represented sets of primitives. In order to represent their borders several edges and faces will be handled as constraints. Updating a topographic dataset therefore equals the addition and removal of constraints within the network. One of the biggest challenges in realization of such a data structure is to reach acceptable performance despite the potential enormous amount of data. The presentation will give insight in the developed model and the biggest challenges for the computational geometry field will be identified.

6. **PCRaster developments, 3D map algebra and error propagation**
16:15-17.00 Derek Karssenberg (UU)

Environmental modelling languages such as PCRaster (http://pcraster.geo.uu.nl) are programming languages embedded in GIS to simulate environmental processes. These languages are used to construct dynamic models, also called forward models, which are simulations run forward in time, where the state of the model at time t is defined as a function of its state in a time step preceding t. Nowadays, environmental modelling languages can deal with forward simulations in two spatial dimensions only, which imposes restrictions on their application field. For future applications, at least two extensions to the languages are required. First, modelling languages need to come with data structures and functions representing three spatial dimensions, since many environmental processes involve spatial interactions in three spatial dimensions. Second, environmental modelling languages need to include Monte Carlo simulation techniques to calculate how input errors propagate to the output(s) of a model. These additions to existing environmental modelling languages require major changes in the syntax, computational engine and visualisation routines of the language, since two dimensions, a spatial dimension and a dimension to represent Monte Carlo loops, are added. In this presentation, I will explain the approach followed by the PCRaster development team to extend their modelling language to modelling in three spatial dimensions and Monte Carlo simulation, and I will discuss some challenges for future research.

———

12-9-2005
www.ncg.knaw.nl