# Spatial Data Algorithm Extension To TRAKLA2 Environment

Jussi Nikander
Department of Computer Science and Engineering
Helsinki University of Technology
Espoo, Finland
jtn@cs.hut.fi

## ABSTRACT

In this paper an extension that brings spatial data algorithm support to the TRAKLA2 system is described. The different exercise types and the problems implementing them are discussed, and suggestions for future work and ideas for better spatial data support to the system are given.

## Keywords

Spatial Data Algorithms, Geometric Algorithms, Teaching, TRAKLA2, Automatic Assessment

## 1. INTRODUCTION

TRAKLA2 [4] is an automated assessment system for teaching data structures and algorithms. The TRAKLA2 exercises are *visual algorithm simulation* exercises, where a group of data structures are illustrated to the user using *algorithm visualization* techniques. In visual algorithm simulation the user manipulates graphical representations with a mouse, thus simulating the actions of an algorithm. Typically the graphics are used to represent data structures. Most exercises in the TRAKLA2 system are *tracing exercises* where the user traces the execution of a given algorithm by manipulating graphical representations of data structures [3]. A typical modification in a tracing exercise is to move a data element from one data structure to another, or from one position in a given structure to another position in the same structure.

TRAKLA2 has so far covered only basic data structures and algorithms. This work describes a current project that aims to extend the idea of visual algorithm simulation to another area of algorithmics. This first extension to TRAKLA2 covers *spatial data algorithms* (SDA) in the field of geoinformatics. Geoinformatics is a branch of science that applies computer science techniques to cartography and other geosciences. Spatial data is a term that covers all data that is located in a multidimensional space. Spatial data algorithms are, therefore, algorithms that manipulate this situated data.

The SDA extension to the TRAKLA2 system covers the basic spatial data algorithms taught to geoinformatics students at Helsinki University of Technology, and is designed to be used on the SDA course arranged by the Institute of Cartography and Geoinformatics at the university. Many of the algorithms taught at the course may be better know to computer scientists as geometric algorithms.

## 2. SPATIAL DATA ALGORITHMS

In geoinformatics, spatial data algorithms are used to manipulate located, geographic data. Typically the data forms some type of map. Map data can be represented using two models: vector and raster. In the vector model, the different terrain features are represented as points, lines, or polygons that have location data associated to them. In the raster model, a regular grid is used to divide the represented area into equal sized cells. Each cell then has a data value associated to it. Spatial data algorithms can be divided into two categories according to which data model they use.

In addition to the two data models and the associated algorithms, there are also several data structures designed to store spatial data. Many such structures can, at least with minor modifications, be used to store data of either model. These structures are typically based on data structures designed to store one-dimensional data such as numbers or strings. For example, an R-Tree [2] is two-dimensional variation of the B-Tree.

### 2.1 Algorithms for Vector Model

In the vector model, the data elements are discrete, and in many cases cover only part of the whole space being examined. Such data can be gathered, for example, by taking samples in the field, and measuring the required data values at these sampling points. Such data is typically modeled using mathematical models, like Voronoi-diagrams [6]. Since the data covers only small part of the area, interpolation and extrapolation methods are important for achieving required coverage.

It is easy to create tracing exercises for most vector model algorithms. For example, in the expanding wave method for creating a TIN-model (a dual of the Voronoi-diagram) [5], the placement of the input points decides the order in which the data points are covered and therefore the execution of the algorithm. In such an exercise, the learner must, after a data point has been covered, decide which point to handle next based on the given input data.

Some of the data structures and algorithms used for handling spatial data in the vector model also demonstrate well-known problem solving methods that can be applied in other fields. The set of implemented exercises includes, for example, finding line segment intersections using the well-known line sweep problem solving strategy.

A total of 12 TRAKLA2 exercises that contain data structures and algorithms for the vector model have been specified for the SDA course. These exercises cover areas such as different structures for storing Voronoi-diagrams, methods for constructing, modifying and analyzing Voronois, methods for traversing and analyzing polygon maps, interpolation methods, etc..

## 2.2 Algorithms for the Raster Model

In the raster model, the whole area being examined is covered by a raster. Such data can be gathered, for example, by digitizing existing maps or from aerial photographs. The data is stored as a matrix of values.

Most of the algorithms used for the raster model include a lot of mathematical calculations. Typical operations are, for example, combining information from two raster layers to a new layer, or generalizing information in one layer. Such operations can be effectively implemented by using filter functions that calculate the values for a new raster layer based on existing values.

In most TRAKLA2 exercises the user traces the execution of an algorithm by moving data elements from one position into another. In most raster problems, however, the algorithm reads the input and uses a mathematical function to calculate the output. Such operations can not be simulated using tracing, but would require other kind of exercises.

Implementing most raster model problems in TRAKLA2 system would require considerable extra work to modify and expand the user interface. These expansions to the system have currently been left as future work. Therefore, only one exercise specific to the raster model has been specified for TRAKLA2.

## 2.3 Visualization of Multidimensional Data

Typically, in TRAKLA2, the data structures are visualized in a way that explicitly shows the internal hierarchy of the structure. A tree, for example, is visualized using a view where nodes are arranged to layers based on their distance from the root node, and each node has links to its child nodes. Root is on the topmost level. Such visualization shows accurately how the data is stored in the data structure. However, if the data is multidimensional, such visualization cannot accurately show the relationships (i.e. distance, direction, size) between different data elements in the given space.

In order to show the relationships between different data elements, a view that shows the given space is required. Typically a two-dimensional space is visualized as an area, where the different data elements are drawn to their correct coordinates. However, such view does not contain information necessary for understanding how the data is stored in the structure. Therefore, in the spatial data algorithm exercises, there is typically a need to show two views of the same data structure simultaneously: one to show how the data is stored and another to show how data elements are related.

## 3. CONCLUSIONS AND FUTURE WORK

Tracing exercises are meaningful when the input of the algorithm affects its control flow. In such exercises the input data must be used to decide what the next step of the algorithm is, forcing the learner to think how to execute the next step of the algorithm simulation. Therefore, in order to solve the exercise correctly, he must know how the algorithm works, and be able to solve the different special cases he encounters. Conversely, if the input has not effect on the control flow of the algorithm, a tracing exercise should not be used. When the steps of the algorithm are the same on each input, the solution process becomes a systematic task, where the learner does not need to think about the next step once he knows how the algorithm works. This, in turn, makes the exercise boring and repetitive.

Knowledge of many vector algorithms can be tested using tracing exercises. However, most raster algorithms are best suited for some other type of exercise. Knowledge of raster algorithms can be tested, for example, by having the user construct a desired filter function. Such exercises are traditionally done either with pen and paper, or by using mathematical programs such as Matlab. However, trying to expand visual algorithm simulation to cover such *exploratory* exercises [3] is an interesting research problem. One possible solution is to borrow ideas of visual programming environments such as Alice [1], and create exercises where the goal is to graphically construct a correct filter function using a given input and a set of possible operations.

## 4. REFERENCES

[1] S. Cooper, W. Dann, and R. Pausch. Teaching objects-first in introductory computer science. *SIGCSE Bulletin*, 35(1):191–195, 2003.

[2] A. Guttman. R-trees: a dynamic index structure for spatial searching. In *SIGMOD '84: Proceedings of the 1984 ACM SIGMOD international conference on Management of data*, pages 47–57, New York, NY, USA, 1984. ACM Press.

[3] A. Korhonen and L. Malmi. Taxonomy of visual algorithm simulation exercises. In *Proceedings of the Third Program Visualization Workshop*, pages 118–125, The University of Warwick, UK, July 2004.

[4] L. Malmi, V. Karavirta, A. Korhonen, J. Nikander, O. Seppälä, and P. Silvasti. Visual algorithm simulation exercise system with automatic assessment: TRAKLA2. *Informatics in Education*, 3(2):267 – 288, 2004.

[5] M. J. McCullagh and C. G. Ross. Delaunay triangulation of a random data set for isarithmic mapping. *The Cartographic Journal*, 17(2), December 1980.

[6] A. Okabe, B. Boots, K. Sugihara, and S. N. Chiu. *Spatial Tessellations: Concepts and Applications of Voronoi Diagrams*. John Wiley & Sons, 2000.