Development of a tool for proximity applications

Anders Dahlgren[1] and Lars Harrie[2]
[1]The National Rural Development Agency, Östersund, Sweden,
anders.dahlgren@glesbygdsverket.se
[2]GIS Centre, Lund University, Lund, Sweden, lars.harrie@nateko.lth.se

MAJOR THEME: Applications of GIS
NATURE OF THE ABSTRACT: Strategic

Abstract

Performing detailed proximity analyses in a national perspective implies dealing with large source datasets. In this paper we present an ongoing work of developing a proximity tool specially designed for handling large dataset. The main focus in the paper is methods for improving the performance of the network search.

KEYWORDS: *network search, generalization of graphs, proximity analysis*

1. INTRODUCTION

Proximity analyses are of interest within many areas; they make an important foundation in describing accessibility relations. In temporal studies they can be used to monitor the change in accessibility over time. Some of these analyses only becomes of real interest when dealing with large geographic areas and/or high resolution in the analyses. This implies that large datasets are required. A problem is that several (on-the-shelf) proximity tools cannot handle large datasets efficiently.

Proximity analyses are of interest for monitoring the situation for the inhabitants in the rural areas. In Sweden, Glesbygdsverket (the National Rural Development Agency) is responsible to define the rural areas and analyze the living conditions for the people living there. Glesbygdsverket uses proximity analyses, for example, to monitor the population’s accessibility to food stores, train stations and hospitals. Glesbygdsverket has a national responsibility; this means that the analyses are performed for the whole country. And because the analyzes require high resolution the datasets inevitable become large. Since Glesbygdsverket did not find a suitable on-the-shelf proximity tool for such large dataset, it has decided to develop its own tool. This paper describes the ongoing work of the development of the proximity tool.

The remaining part of the paper is structured as follows. In section 2, a short description of the current proximity tool is given and section 3 is devoted to requirements of the new tool. In section 4, possible improvements of the new tool are discussed with focus on network search. The paper ends with concluding remarks.

2. CURRENT TOOL FOR PROXIMITY ANALYSIS

Glesbygdsverket has developed its own proximity tool. The first generation of the tool was developed in the early nineties. When the requirement of the analyses increased, as well as the size of the datasets, the performance of the calculations became a growing problem. In the years 2002–2003, a second generation of the tool was developed. This version is referenced in this paper as *MapProx* (*Map* for the mapping capabilities and *Prox* because it is a proximity tool). An example of the output from *MapProx* is shown in figure 1.

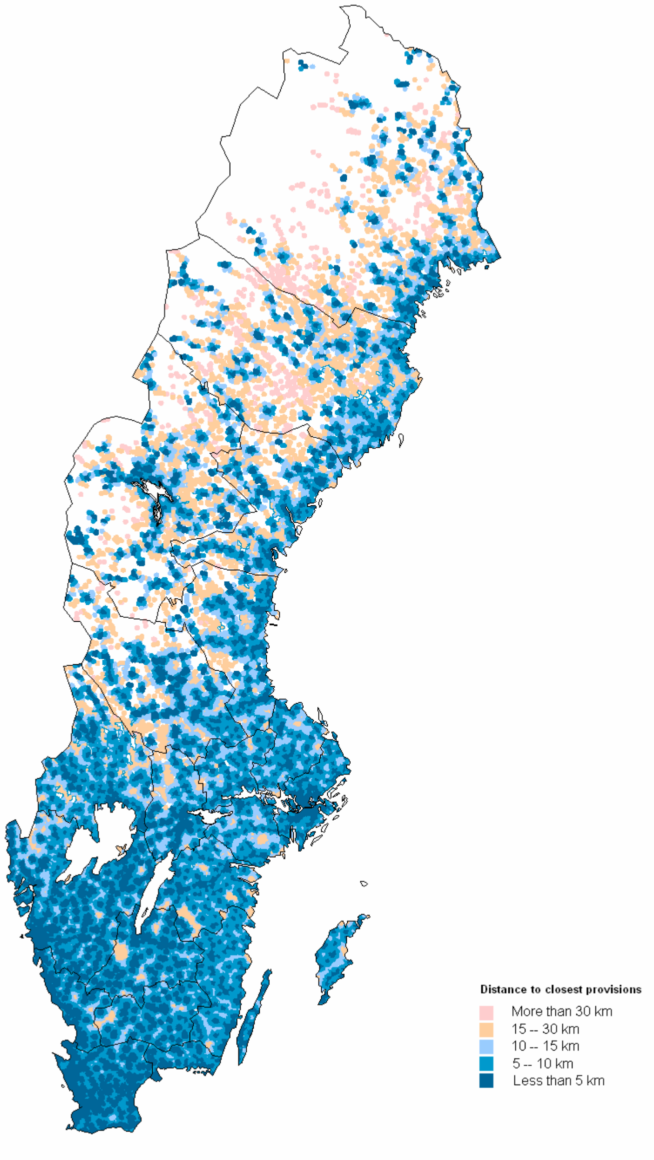

Figure 1: A thematic map showing an example of the output Glesbygdsverket's proximity application *MapProx* showing the Swedish populations (2004) accessibility to the closest grocery store. Glesbygdsverket (2007).

3. REQUIREMENTS ON THE NEW TOOL FOR PROXIMITY ANALYSIS

The design of the new tool should have a scalable approach. The source datasets used within Glesbygdsverket will certainty become larger, both expanding into new geographic areas (foremost Europe) and using higher resolution. Examples of the latter are:

- To use the national road dataset (NVDB) from the Swedish Road Administration instead of the roads from the 100 000 topographic map (*Blå kartan*) from the Swedish National Land Survey. This implies an increase of the number of roads with a factor of two.
- To use a population grid of a resolution of 250 m instead of 1 km. There are about 120 000 populated km-squares and about 400 000 populated 250m-squares in Sweden.

At a first glance the process of performing proximity analysis seems to be a fairly static task: *once it's done it's done*. But considering changes in the source datasets and simulation functions dealing with iterations, means that a lot of calculations are required. This makes high performance an important requirement on the application.

The development of the application should be performed in such a way that dependencies to operating systems and GIS-platforms are held to a minimum. This allows the application to evolve into heterogeneous computer environments.

The design of the application shall consider the future trends in the computer hardware development aiming at e.g. computers with multiple processors and 64-bits processors.

A requirement that of course is of interest is the applications usability. When doing time-consuming calculations it is important for the user to get information on the progress. This also has an impact on the performance while updating the user interface uses computer power.

When new source dataset is used in the application, there are often small changes from the datasets used in earlier calculations. In *MapProx* today a total recalculation is made regardless of the size or type of the changes. If some incremental update routines could be implemented they will with all certainty a positive effect on performance.

The application should have the possibility to use a categorization of the target points. This function can then be used to calculate the shortest distance to more than one type of target points in the same calculation.

The *MapProx* tool has two implementations. Firstly it will work as a desktop application for the employees of Glesbygdsverket, doing straightforward proximity analyses. Secondly, it will work as a module in larger, more complex systems that have proximity analyses as a base function. Examples of the latter are program systems for rescue planning and for calculating tax adjustments between municipalities[1]. This means that the interface to the functions in the tool could be accessed of both a human and another computer process.

When a new system is developed or bought by an organization the transparency of the system should be considered. Buying a ready made system in most cases tend to become a "black box" with few or no possibilities changing the system according to the requirements of the organization, whereas a system developed inside the organization becomes a "transparent box". It seems to be a growing requirement from organizations to get access to the source code for the applications they use. This is a point often brought forward in discussions in favor of Open Source concepts (Raymond, 2001). Doing proximity analysis is a core function within the Glesbygdsverket, which makes it important to have a clear insight into the details of the application. Glesbygdsverket wants to have access to the source code to have this insight. Of course this discussion of trying to rank, developing a new system with buying an on-the-shelf system, is more complex..

---

[1] Glesbygdsverket assists the Ministry of Finance to calculate tax adjustments dependent on structural differences between municipalities.

4. IDENTIFYING RESEARCH AREAS

A performance test was made in Dahlgren (2005) where four candidates for performance improving research areas are selected. Three of the areas can be directly related to the measurable sub processes in the test:

- Create a spatial index and connect the points to the road network
- Generalize the road network with the connected points.
- Calculate the shortest distance between start and target points.

The overall interest area of the internal data structure can be added to this list. This area is affected of all the three above but should because of its importance be handled separately. The points identified above delimit this work. There are of course other areas that could affect the performance of the application and should be treated in the development project but they are not the concern of this paper.

4.1 Create a spatial index and connect the points to the road network

This process has been previously studied in Dahlgren and Harrie (2006) and will not be described here.

4.2 Generalization of the road network with the connected points.

The *MapProx* tool is a one-purpose tool, namely performing proximity calculations. Tools with similar functions often have a multipurpose approach. They often tend to broaden their field of operation to please a vaste amount of users. However, such designs often lead to compromises in the core function of the application. An example of this is the mixing of proximity analysis and navigation. In a proximity analysis tool you are not interested in a description of the path from start point to target point. You just want to know the travel distance or travel time. This fact has impact on the generalization process where a pure proximity calculation can use a more effective approach. Removing relations between the complete geographic representation of the road network and the generalized road network is no problem.

Start points and target points must be connected to the road network before the generalization process can begin. The goal with the process is to simplify the road network to make the calculations of the distances as smooth as possible without tempering with the correctness of the result.

4.3 Calculate the shortest distances between start and target points in the generalized network

Finding the shortest way in a network graph is a classic problem in computer science and mathematics. The algorithm used in the tested application is the Dijkstra algorithm (Dijkstra, 1959) later described and discussed in many sources (e.g. de Berg et al, 1997; Ahuja et al., 1993). There have also been a number of suggestions of improvements and modifications of the Dijkstra algorithm; e.g. by using heuristics as in Lauther (2004) and Ertl (1998). Dijkstra's algorithm has been proven to be the fastest exact search algorithm but heuristics could be of interest for the further development of *MapProx*. A test in Dahlgren (2005) showed an increased calculation time with increased search distances. The recommendation is still to prioritize the internal data structure before going into details of the search algorithm.

4.4 The internal data structure

Figure 2 shows the internal format in the current application uses in a simplified manner. There is a geographic area tiled into of sub areas. One sub area tile holds the reference to the corresponding links and nodes that lay within the area. The advantage with the approach of tiling the geographic area into sub areas is that the sub areas become "independent" from each other, which make them easier to handle. The Sub areas are serialized into individual files in the internal format. The sub areas can be handled independently in their own calculating threads, suggesting parallel calculations as a possible performance enhancing measure. The disadvantage with the tiling is that a new type of nodes must be introduced on the borderlines between the tiles. These border-nodes will appear in two adjacent sub tiles. The intimate connections between links and nodes (in the implementation realized with pointers) are the core of the internal topology of the internal format. As long as the road network stays intact the same imported network can be reused in forthcoming calculations.

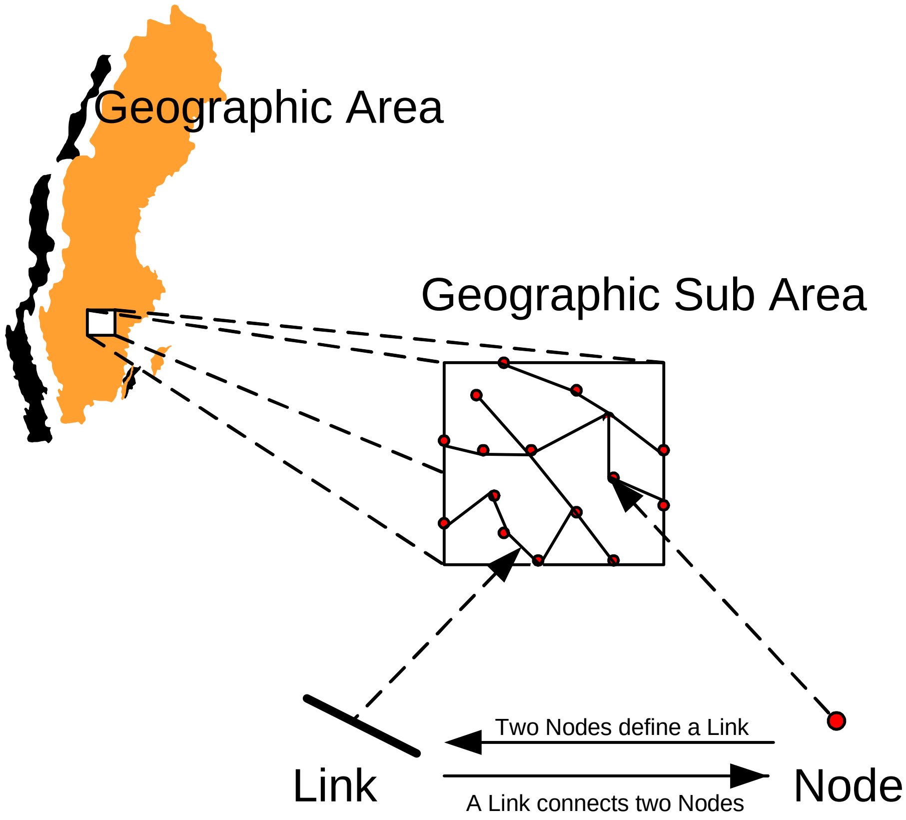

Figure 2. The figure describes a simplified schema of the internal format used.

A suggestion is to let the geographic sub area have a polymorphic behavior. The area should have three different interfaces, illustrated in figure 3, used as described below. It remains to be tested if this way of structure of the data has an effect of performance or only works as a structural clarification.

The processes that connect start and target points to the road network use the first interface. In this interface it is important that the right geographic representation of the road network is used. When the start and target points are connected the generalization process can start presenting its result, a simplified road network, through the second interface. This is used for searching within a Geographic sub area. When a sub area only has the function as transportation between two other sub areas, i.e. the area does not have any start or target point of interest for the calculation connected, the third interface is used. This interface shows the sub area as a matrix of distances between the sub areas border nodes.

An interesting fact is that the matrix can be calculated before the connection of the start and target nodes in a precalculation.

In the design of the internal format it should be if there are processes that could use parallel calculations. One overall calculation process can use both a single thread approach in a sub calculation and a multithread approach in another (figure 5). Examples of this can be the import of a single in-data file that is a single thread procedure. When the dataset is imported and tiled into subsets, a multithread approach can be used to calculate the matrix representation. The tests in Dahlgren 2005 could not directly pinpoint any bottlenecks in the internal data structure because it affects the whole calculation process. When the size of the datasets increases a hierarchical approach (Car and Frank, 1994) and the suggested interface structure should be of interest of study further.

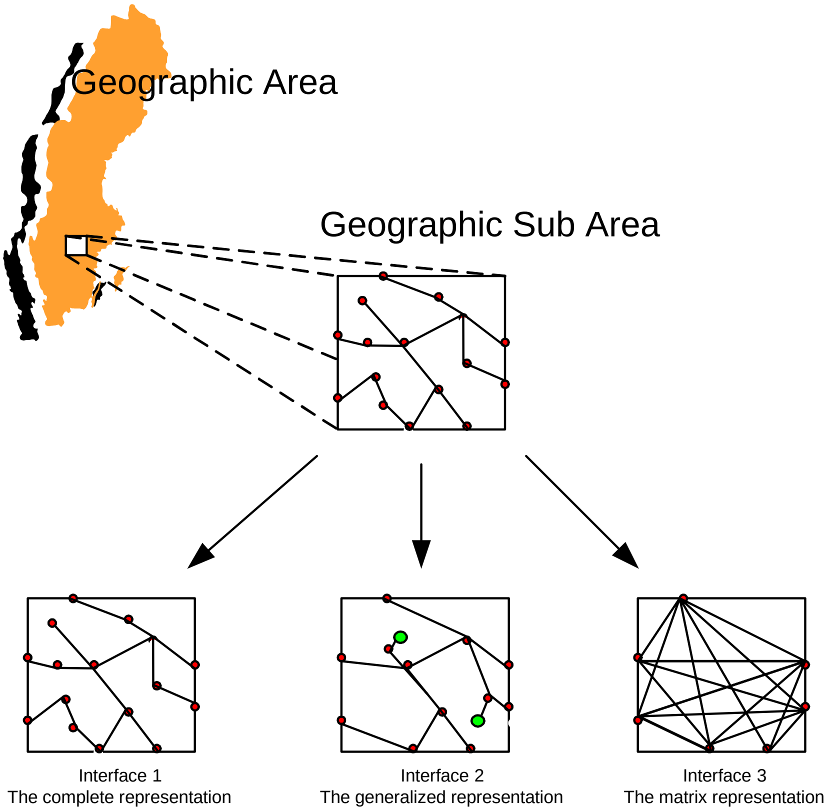

Figure 3. Three interfaces of the Geographic Sub Area component. The big nodes in Interface 2 represents start or target nodes.

In Figure 2 & 3 the geographic sub areas are illustrated as squares. In reality this may not be the optimal way of tiling the road network. A method of expanding rural areas with dense road networks into geographic sub areas is suggested (figure 4). At least two advantages with this method can be seen. Firstly the probability of internal proximity analysis within the sub area increases. It is often the tool is used for finding the closest target point in the form of service within rural areas. Secondly the introduced borders between the geographic sub areas are drawn in rural areas with a sparse road network. Suggesting a low number of border nodes to be applied.

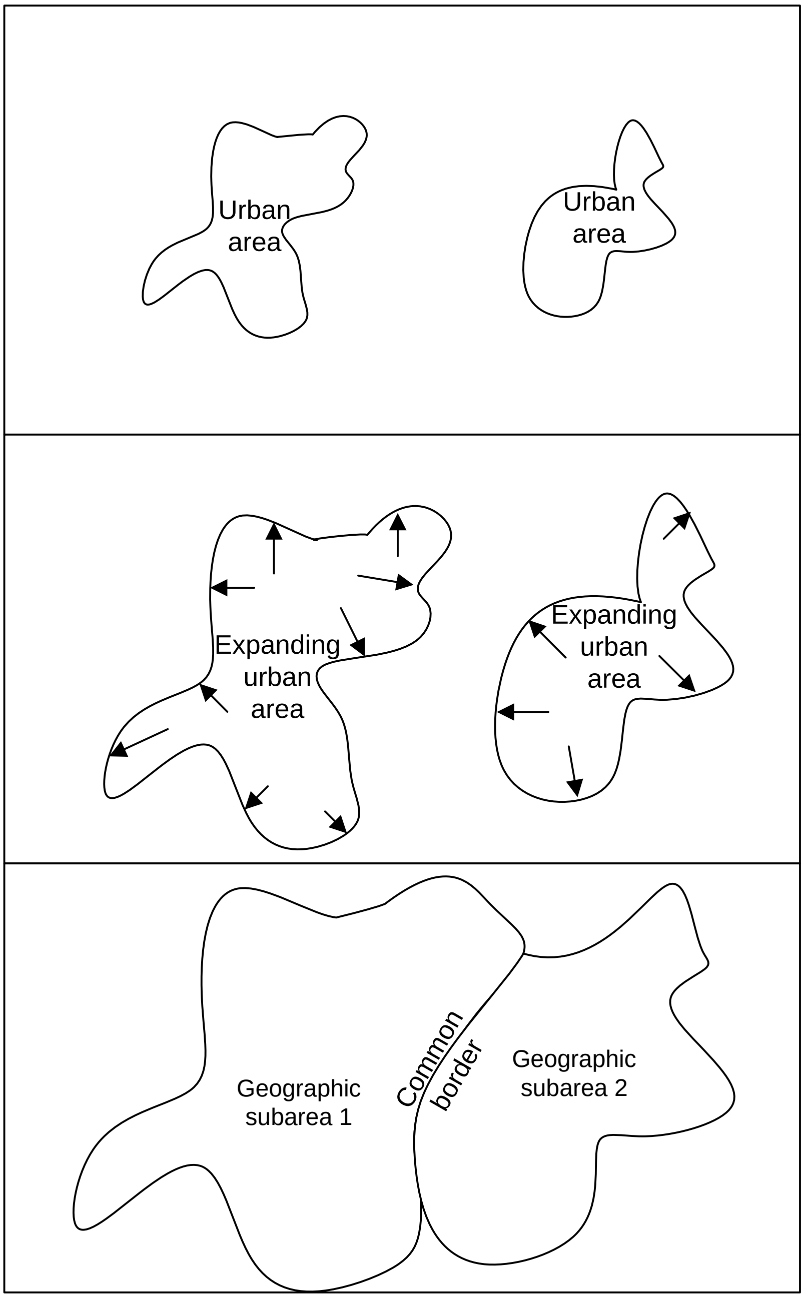

Figure 4. A method of constructing Geographic subareas by expanding urban areas.

5. CONCLUDING REMARKS

This study describes an ongoing project of developing a new version of a proximity tool (*MapProx*). The requirements stated above are the guidelines of the development work. The focus in this development project is to improve the performance of the application. This is done by going through the different calculation steps and suggesting improvements. An area that affects all calculation steps is the internal data structure which is the focus of this paper. The internal data structure is the concern of all the calculation steps and can therefore not be looked at as an isolated problem.

ACKNOWLEDGEMENTS

We would like to thank Professor Anders Östman for his support and the Swedish National Rural Development Agency and the Swedish Rescue Services Agency for financing the studies.

BIBLIOGRAPHY

Ahuja, R.K., J.B. Orlin, and T.L. Magnanti, 1993. *Network flows: theory, algorithms and applications*. Englewood Cliffs, N.J.: Prentice Hall. xv, 846.

de Berg, M., M. van Kreveld, M. Overmars, and O. Schwarzkopf, 1997. *Computational Geometry: Algorithms and Applications*, Springer.

Car, A., and A. U. Frank, . and General Principles of Hierarchical Spatial Reasoning - The Case of Wayfinding. *Proceedings of the 6th Spatial Data Handling Symposium*, Edinburgh, 1994. Vol. 2: p. pp. 646-664.

Dahlgren, A., 2005. A proximity analysis application for large source dataset. *Proceedings of ScanGIS 2005*, Stockholm, pp. 125-136.

Dahlgren, A., and L. Harrie, 2006. Evaluation of computational methods for connecting points to large networks, *Mapping and Image Science,* No. 4 , pp. 45-54.

Dijkstra, E.W., A note on two problems in connection with graphs. *Nummer. Math*., 1959. 1: p. pp. 215-248.

Ertl, G., Shortest path calculation in large road networks. *OR Spektrum* 20, 1998: p. 15-20.

Glesbygdsverket, 2007, The National Rural Development Agency´s Homepage, www.glesbygdsverket.se.

Lauther, U., An Extremely fast, exact algorithm for finding shortest paths in static networks with geographical background. *Geoinformation und Mobilität*, 2004. 22: p. 219-230.

Raymond, E.S., *The cathedral and the bazaar: musings on Linux and open source by an accidental revolutionary*. Rev. ed. 2001, Sebastopol, Calif.: O'Reilly. 241.