**Integrating heuristics and spatial databases: a case study**

*Cid Carvalho de Souza*
*Claudia Bauzer Medeiros*
*Ricardo Scachetti Pereira*
{*cid cmbm scachett*}*@dcc.unicamp.br*

**Relatório Técnico IC–96-18**

Dezembro de 1996

# Integrating heuristics and spatial databases: a case study

Cid Carvalho de Souza  Claudia Bauzer Medeiros
Ricardo Scachetti Pereira
{cid cmbm scachett}@dcc.unicamp.br
IC - UNICAMP - CP 6176
13081-970 Campinas - SP Brazil

**Abstract**

This paper presents part of the ongoing efforts at IC-UNICAMP to apply heuristic algorithms to vectorial georeferenced data in order to help decision support in urban planning. The results reported are original in the sense that they combine recent research in both combinatorial algorithm development and geographic databases, using them in the solution of a practical problem. A first prototype, described in the paper, has already been developed and tested against real data on the city of Campinas, to support planning activities for the São Paulo State Post Office System, Brazil.

**Keywords:** Heuristics, spatial databases, urban planning.

## 1 Introduction

GIS are becoming a very important means for decision support in several domains, ranging from environmental to urban applications in different scales and from distinct perspectives. In an urban context, they are being adopted by both private enterprises (e.g., for selecting a shopping center site) and public services (e.g., for determining location of health or education facilities).

In spite of such growing demand, existing solutions do not take advantage of recent research results in multicriteria analysis, combinatorics and optimization (C & O), and operations research. In fact, the integration of optimization heuristics to GIS in order to support decision taking is still in its infancy. Furthermore, C & O experts usually ignore the advantages of using GIS to help visualization and manipulation of spatial data. Thus, though important results are being obtained, on one hand, in development of GIS applications that require C & O expertise, and, on the other hand, in the application of C & O techniques to decision support, there is a gap between research in these fields.

Some commercial GIS begun to advertise tools that address this issue but at a primitive level, making it very hard to tailor solutions to a given spatio-temporal context. Like many problems involving spatial data, solutions for a particular region cannot be generalized, due to the need for adjustment to regional characteristics. Furthermore, several GIS based decision support solutions developed in some countries (notably in the USA and parts of Europe) cannot be applied to the Brazilian scenario, given the expansion pattern of Brazilian urban centers. Such centers are characterized by intensive urban growth and frequent urban migration, causing constant and significant changes in city map topography. Thus, an effort must be made to provide flexible optimization tools to GIS users, which take advantage of C & O results.

This paper describes some of the results of an ongoing project whose aim is to reduce this gap between research and development in C & O techniques and GIS technology for urban planning applications. The main contribution reported lies in emphasizing the issues that must be taken into consideration in order to increase computation flexibility in current GIS. This, in turn, may help designing systems for a large family of applications which require innovative solutions combining optimization, spatial databases and cartographic visualization.

The rest of this paper is structured as follows. Section 2 gives an overview of our case study – planning urban postal services in Brazil – and describes in detail the problem which we have solved using a combination of optimization heuristics and spatial databases. Section 3 describes the solution, from an algorithmic approach. Section 4 describes the prototype developed and shows some examples ran against data for the city of Campinas (population 900.000). Section 5 provides a brief comparison to related work. Finally, section 6 contains conclusions and describes the current stage of this project.

# 2 Overview of the problem

## 2.1 Overall description

We concentrated on the development of flexible decision support tools for planning mail distribution in urban areas in the state of São Paulo (population: roughly 30 million). Planning goes from the macro (state) level to the micro (street distribution) scale. GIS can bring considerable impact to organize tasks including routing, scheduling, dispatching, transportation logistics and others.

There is a wide variation in parameters associated with mail distribution in a given urban area. Some relevant spatio-temporal issues in São Paulo state are:

- Non-uniform (spatial) demand for mail services.

  There is not always a direct correlation between population density (or, for that matter, spatial extension) and mail volume. Commercial zones, even when they occupy a small spatial extension, may have a high concentration of what the postal service calls "large users", e.g., recipients of large volumes of correspondence, typically commercial or govenment offices, located at a single address. On the other hand, some large areas, even when densely populated, receive comparatively less mail.

- Spatio-temporal dependence.

  Mail volume has a spatio-temporal fluctuation: affluent residential sections often show a marked decrease in correspondence volume during summer (long) holiday vacations, whereas summer resorts present an increase in volume during the same period, causing workload imbalance.

- Spatio-temporal evolution of street topology and population density.

  This is a very important issue in developing countries such as Brazil, where large urban centers present a marked migration pattern. In the case of cities like Campinas, for example, migration from other states is causing a major impact in postal services. New streets may be created, sometimes even in a monthly basis, which requires constant updates and replanning of mail distribution tasks.

- Social issues

  Though these will not be considered in this paper, distribution planning should also take into consideration particularities in mailmen social habits (e.g., a given mailman may never follow his scheduled itinerary because he may want to stop for a coffee at a certain time in his favorite bar). Thus, several planned schedules actually fail because the people involved do not execute the tasks defined.

All these issues increase the difficulty in maintaining an updated database (both spatial and non-spatial).

### 2.2 Spatio-temporal data

Data used in mail distribution planning comprises spatial data about the region considered; temporal/seasonal data about correspondence volume, on a weekly basis, for the region; and statistics on average delivery time taken (or estimated) by mailmen. Spatial data is obtained by map digitalization, stored in vector format[1]. Nonspatial data is stored in a relational database.

The basis for spatial data is a set of city maps, where streets are represented according to the *centerline* paradigm (i.e., a street is abstracted by a polyline, where segments are usually determined by street intersections, and street width is of no importance). A map is represented as a network of centerline segments. Buildings are placed with respect to this network, being positioned on either side of the centerline which represents the street where they are located. Side definition is important since distinct mailmen may work on different sides of the same street. At each level, the routing and clustering problems present their own particularities, and data management is performed in different ways. Data is used in several levels of detail, for distinct purposes – building, district, distribution zone, distribution unit:

- At the largest scale, spatial data includes *individual buildings* (houses, apartment blocks). Non-spatial data, in this context, describes every mail recipient, by address. For mail distribution purposes, however, only individual building locations are important.

[1] Digital maps are presently not available at the post office, being provided by some municipalities.

- At a smaller scale, street segments are grouped into *districts*, where a district is the mail delivery area which is under the responsability of one mailman. A district is thus a street network describing one mailman's daily delivery route (district and mailman may be considered synonims as far as space-related work units are considered). *District determination* may partition a given street into several segments, each of which allocated to one mailman.

- The next level groups districts in terms of *distribution zones*, partitioning the map into areas which cover several districts. Each zone contains one site (the distribution center) which centralizes correspondence distribution for all districts within it. Mailmen are assigned to a distribution center and start their daily work by going to this site to get correspondence. One big city may have several distribution centers.

- Finally, at the highest level, distribution zones are aggregated into *main distribution units*, which correspond to main mail routing foci, usually cities. In sparsely populated regions, a set of cities may be represented by a single node.

Textual data consists essentially of temporal statistics on mail volume and mailmen average working time. One complicating factor from a database point of view is that statistics are defined in terms of a spatial unit that is internal to the post office, here called *street delivery unit* (SDU), and which is not compatible to the spatial (centerline) database. A SDU is a centerline polyline whose end nodes correspond to street corners and which measures ideally 600m. It is usually specified as: "the segment of street X that goes from the intersection of street K to the intersection of street J". In a city map, this polyline may correspond to several small streets linked together, or part of a long avenue, or even just one side of part of a street. One statistics record provides information like: "in SDU $S$, delivery time is 25min from november to january". This means that our system must perform a nontrivial spatial data conversion in order to link statistics and street maps.

Today, the definition of distribution nodes and zones is done manually. The determination of the streets (and buildings) that form a district is done semi-manually. A few optimization procedures provide preliminary parameters, and the actual district determination is done manually by teams of experts, usually on paper maps.

As an example of the work involved, the definition of the streets for the districts in a single distribution zone in the city of São Paulo (population 12 million), may take as much as one week of work for a team of 3 people working full time (i.e., 120 man hours). This expenditure of time is mainly due to two main reasons. First, all allocation decisions are manual, and based on crossing paper maps against statistics data, for a given region. Second, for areas in the outskirts of the city, there are constant updates due to urban growth, and these updates consume much time since data is spread in different data files.

In fact, one of the main obstacles for using GIS in this context is the amount of data sources and formats, as well as that, in large urban areas, a very high rate of urban modifications (constructions, street openings etc) may happen without being recorded in municipality maps (being only known to the mailmen that work in the area). District determination is a priority goal for automation, and the prototype described in section 4 was developed in response to this demand. From now on, therefore, we will concentrate on the description of this specific problem.

### 2.3 District determination in an urban area

District determination corresponds to specifying a (connected) street network which will be traversed by one mailman. District determination has two main goals: to minimize the number of mailmen working in a distribution zone; and to evenly balance, among these mailmen, the daily work load of mail delivery, within this distribution zone. This load is defined as having an upper limit of 480 minutes per person/day (8 hours). This is the goal of our optimization and includes the displacement time of each mailman from the distribution center to the start point of his district (i.e., the time it takes a mailman to reach the place from which to start mail delivery).

Average delivery times are expressed in terms of SDU spatial units. Usually an SDU corresponds to a street, or parts thereof. There are, however, exceptions: some streets have different delivery units computed for each side (and thus must be processed twice); others may have an exceptional high or low volume of correspondence (and thus break the 600m spatial constraint for an SDU specification). Another factor to consider is the type of mail received in a given SDU: regular letters, express packages and so on. All such variations are computed and counted for each SDU over a period of time, in order to obtain the final statistics. Furthermore, for some zones, there are distinct statistics records for different seasons.

Finally, the topography of a region is also considered in order to increase or decrease average delivery parameters. For instance, in residential areas with little incidence of slopes, the average delivery time on foot is 5.74 km/h, whereas commercial areas in hilly zones average 4.41 km/h. Speed, of course, also varies according to the means of mail delivery: pedestrian areas or highways must be treated as exceptions.

## 3 Description of the solution – combining heuristics and spatial databases

This section describes the heuristics we have developed in our first prototype to tackle the problem of *district determination*. We modelled the original problem as a graph partitioning problem, where the graph was built by combining the spatial database and statistics data provided by the Post Office. We then used different heuristics to generate solutions which minimize the number of mailmen necessary to ensure the distribution in the zone being considered. In order to combine data sources we had to perform a spatial unit conversion (see section 4).

Though the goals in district determination are more general than simply minimizing the manpower necessary for distribution, one of the central questions for the Postal Service was to know, for a given zone, if the current number of districts (i.e., mailmen) was not overestimated. Moreover, as we shall see later, any solution that attempts to minimize the number of mailmen can be used as a basis for finding a solution that balances the workload among the mailmen. For basic definitions in graph theory we refer to [Ber73].

### 3.1 Graph Partitioning Model

Two constraints are fundamental in defining a district – one nonspatial and the other spatial. The first is the mailman daily load. The second is intrinsically related to the street map and deals with the connectivity of the streets covered by the district. The reason for requiring a district to be connected is that a mailman should not have to go through streets in which he does not deliver any letter. To determine whether or not a given district is connected we have to make use of the spatial database.

We now show how to build a undirected graph $G = (V, E)$ that incorporates all the necessary information for solving **the district determination problem**. First, we transform the zone street map into an SDU map. Next, we associate every SDU to a node in $V$. Moreover, to each node $u \in V$, we assign a weight $t_u$ which is equal to the **average delivery time** of the corresponding SDU. It remains to define when an edge is in $E$.

As mentioned before, the city map is represented according to the *centerline* paradigm. If the two SDU corresponding to nodes $u$ and $v$ in $V$ are such that their polylines share a common point (endpoint of a line segment of both polylines) then, the edge $(u, v)$ is in $E$. In other words, this representation ensures that we can go from the SDU corresponding to node $u$ to that corresponding to node $v$ without having to pass in any other street of the city if and only if $(u, v)$ is in $E$.

Let $W$ be the maximum workload allowed for a mailman in a day (i.e., 480 minutes). Consider a partition $(V_1, V_2, \ldots, V_k)$ of the nodes in $V$ such that: (i) $\sum_{u \in V_j} t_u \leq W$ and (ii) the subgraph induced by $V_j$ is connected for all $j \in \{1, \ldots, k\}$. Denote a partition satisfying (i) and (ii) by *W-connected partition of G*. Clearly, every $W$-connected partition of $G$ corresponds to a valid district partition of the distribution zone considered.

Thus, our first goal is to find a minimum size $W$-connected partition of $G$. Unfortunately, this problem is $\mathcal{NP}$-hard. This can be shown from a simple reduction of the bin packing problem ([GJ79]). Therefore, we tackled the problem using heuristic algorithms.

### 3.2 Heuristics implemented

From now on, the term *cluster* will denote a subset $V_j$ (and its induced subgraph) of a $W$-connected partition of $G$. As noted above, each cluster corresponds to a valid district.

In principle, one could think of two sort of heuristics for this problem: *construction heuristics*, where the solution is built from scratch, building one cluster at a time until all nodes have been assigned; and *local improvement* (search) heuristics, in which several solutions are visited iteratively, using a set of predefined operations to navigate across solutions. Local improvement heuristics assume the existence of a initial partition of $V$ and, in this particular, are suited to the district problem (since the Postal Service already has a district partition of the zone). These heuristics are especially attractive when the goal is to balance the workload. For more on local improvement heuristics, we refer to [PS82].

However, since our initial goal is to minimize the number of clusters in the partition (number of districts in the distribution zone), we ignored the fact that an initial solution was already available.

We implemented two types of algorithms: *simple greedy algorithms*, using heuristics $H1$ and $H2$; and *greedy randomized adaptative* algorithms, using heuristics $H3$ and $H4$. The basic steps of heuristics $H1$ and $H2$ are:

1. Initialize all nodes (SDU) $u$ in $V$ as *NOT ASSIGNED* and set $i \leftarrow 0$.

2. Create a cluster (district) $V_i$ with no nodes (empty set).

3. $r \leftarrow CHOOSE\_ROOT(V_i)$ and mark $r$ as *ASSIGNED*.

4. Let $Q$ be the set of nodes $v$ such that there exists a node $u$ in $V_i$ with $(u, v) \in E$ ($u$ and $v$ are adjacent). Let $Q'$ be the set of nodes $v$ of $Q$ which are *NOT ASSIGNED* and such that the sum of the node weights in $V_i$ plus $t_v$ is no greater than $W$ ($Q'$ = SDUs not assigned to any district and which still fit into the district being computed).

   If $Q'$ is empty, go to step 6. Else, let $v$ be the node in $Q'$ with the smallest number of adjacent nodes which are *NOT ASSIGNED*.

5. $V_i \leftarrow V_i \cup \{v\}$, mark $v$ as *ASSIGNED* and repeat the previous step.

6. If there are nodes *NOT ASSIGNED* in $V$, set $i \leftarrow i + 1$ and go to step 2

The difference between heuristics $H1$ and $H2$ lies in procedure *CHOOSE_ROOT* (step 3). The node used to start building a cluster is called the *root node of a cluster*. In heuristics $H1$, the root of a new cluster is chosen randomly; in $H2$, the root is chosen by picking, among the *NOT ASSIGNED* nodes, the node with the smallest number of *NOT ASSIGNED* adjacent nodes. The choice implemented in $H2$ was more effective, as shown in section 4.

Heuristics $H3$ and $H4$ are based on a modern heuristic paradigm called GRASP (Greed Randomized Adaptative Search Procedure) ([FR95]). The idea of GRASP is quite simple. Its first step generates a solution using a greedy algorithm which does not pick the (local) best element at each iteration. Instead, it randomizes this choice among a set of $k$ best elements. The second step is a local improvement (search) procedure which starts from the solution generated in the first step. These two steps are then repeated $\ell$ times and the heuristic returns the best solution found during this process.

A pseudocode for a GRASP heuristic (for a minimization problem) is the following:

1. $c_{best} \leftarrow \infty$

2. **Repeat** $\ell$ times:

   2.1 $S \leftarrow GREEDY\_RANDOM(G)$.

   2.2 $S \leftarrow LOCAL\_SEARCH(G, S)$.

   2.3 **If** $c(S) < c_{best}$, $c_{best} \leftarrow c(S)$ and $S_{best} \leftarrow S$.

3. **Return** $S_{best}$.

The local improvement phase is not necessary to minimize the number of districts. We developed two modified versions of GRASP. The first one, denoted by $H3$, uses a randomized version of $H1$ in step 2.1 while the other, denoted by $H4$, uses a randomized version of $H2$. For both $H3$ and $H4$, in our tests, $k$ was set to 3 and $\ell$ was set either to 100 or to 500. By setting $\ell$ to 500, the running time of the algorithms for the instances we have tested was always close to one minute, for zones with up to 2000 street segments.

# 4 Implementation

The solution described in section 3 was implemented into a prototype coded in C++ which runs in a Pentium workstation. The base maps used were provided by the municipality of Campinas, and SDU temporal statistics data was provided by the Campinas central Post Office. The prototype presents the results by means of maps, for a given distribution zone, where each district is assigned a different color.

## 4.1 Functional description and architecture

Besides computing solutions to the district determination problem, the prototype allows user interaction in the following ways:

- update non-spatial data;
- change optimization input parameters, in order to compute alternatives to a solution (e.g., changing expected volume of mail for a given period in a given SDU);
- choose the heuristics to use in computing a solution;
- locate and classify districts according to distinct parameters.

Our architecture is based on an open systems' philosophy, and consists of integrating modules by exporting and importing data and performing data conversion routines. This fosters interoperability, incremental development [FK95] and, conceivably, the modules can be used in other decision support environments.

The prototype is composed of three main modules:

- End user interface
- Data structure manipulation module
- Optimization/heuristics function library

These modules were coupled to the spatial and statistics databases (statistics on SDUs is provided by means of relational tables).

First, the user selects the region of work by specifying the distribution zone to be analyzed. Furthermore, the user may provide initial optimization parameters (e.g., changing workload constraints). From this initial specification, the system retrieves from the databases the adequate street network and postal statistics data for the SDU within the zone.

Next, the *data structure manipulation* module processes this data in order to construct intermediate structures that are used by the optimization library. We repeat that one complicated issue from a spatial database point of view is the mismatch between SDU spatial units and the street network in the spatial database. This requires an initial conversion step in which SDU are matched against the street network by means of an operation similar to a spatial join.

Once this join is performed, the data structures described in section 3 are built and fed to the *optimization library*. The user can then iterate through the computation of different solutions, which are presented in two ways: either as a graph or as a map, where the districts computed are outlined by different colors.

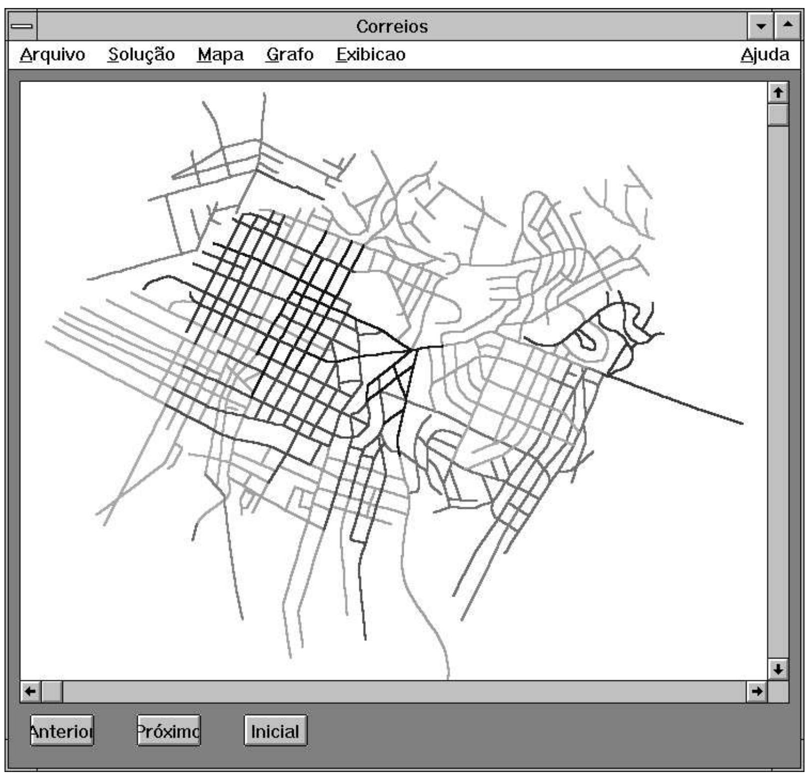


Figure 1: Map showing one district, for the graph in Figure 2

Since the prototype was developed to demonstrate to the Post Office the effectiveness of merging heuristics and spatial databases, many design decisions reflected the need for simplified (but timely) development, rather than what might be desired in a real system. A first implementation simplification was due to the fact that a distribution zone is contained within a polygon whose spatial boundaries are predefined street segments, and cannot be changed (among other things, for political reasons). Ideally, a distribution zone should be retrieved by a window query on the city map (i.e., allowing flexibility in defining the zone), but we simplified the retrieval procedures by storing each zone separately.

A second simplification was the fact that we did not consider the problem of solution presentation (neither in the combinatorial sense of minimizing the number of colors used, nor in the interface sense of output ergonomy). The interface module just assigns a different color to each district in a zone, without considering its visual impact.

Both simplifications correspond to relevant issues from a GIS software development point of view, and will be considered in the next implementation version. The flexibility in distribution zone boundaries must be ensured in the future, in case the postal services decide that their boundaries can be changed. Furthermore, this is needed to support planning in another scale level (that of zone determination within a city).

Figures 1 and 2 show two examples of screen displays taken from the prototype. The user can specify the zone of interest (option *Arquivo*), provide initial parameters and choose different algorithms to compute the solution (option *Solução*), and work either with the map (*Mapa*) or with the graph (*Grafo*) representation. Menu option *Exibição* allows the user to iterate through a set of individual districts. Figure 1 shows the solution for just one district (in bold), within a zone, for the city of Campinas. The user can request a partial solution only (i.e., one district), which can then be interactively modified by changing general constraints or, in some cases, SDU definition. Figure 2 shows the graph generated by this solution.

Showing the graph may be useful in some cases, e.g., in order to identify problems. For instance, the graph in figure 2 shows small clusters of SDU which are not connected to the rest of the graph. This may signal, for example, errors in the spatial database, less visible in the map display (Figure 1).

### 4.2 Computational results

In order to evaluate our solution, we should ideally compare it to optimal results. But, since the problem is $\mathcal{NP}$-hard, an optimal solution is usually not available and the alternative is to make the comparison against some known *lower bound* to the (minimization) problem.

A trivial lower bound for the minimum size of a $W$-connected partition of a graph $G$ is given by rounding up the quotient between the sum of all node weights and $W$. Thus, if $LB$ denotes this lower bound, the following formula holds:

$$LB = \left\lceil \frac{\sum_{u \in V} t_u}{W} \right\rceil .$$

In other words, this expression tells us that at least $LB$ mailmen (districts) are necessary to guarantee the distribution in the region considered.

We ran tests against three distribution zones in the city of Campinas. The data corresponding to the centerline maps of these regions are summarized in the table below (Map 1 is the map shown in Figure 1).

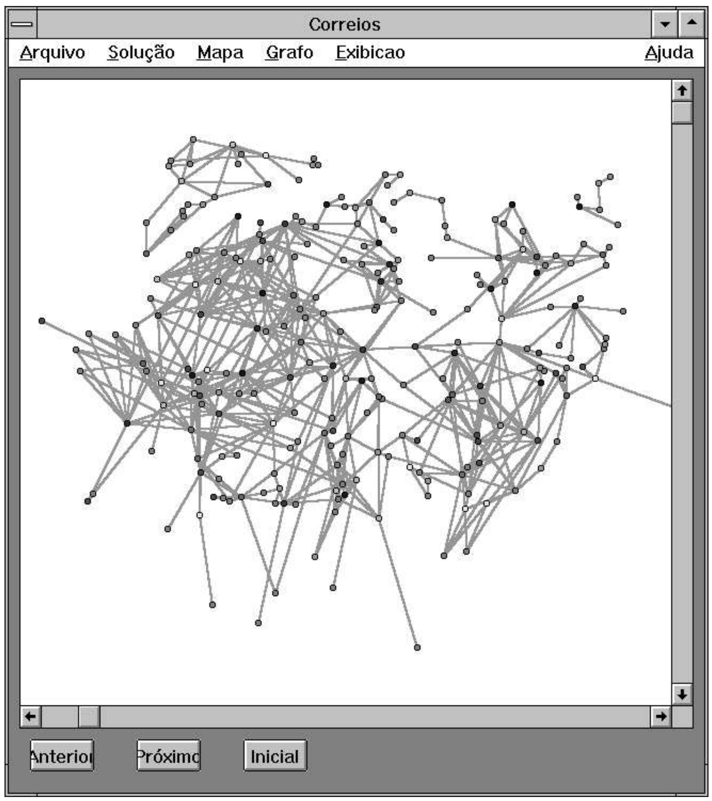


Figure 2: Graph showing solution displayed in Figure 1

| Map id | Points | Street Segments | SDU | Solution avg time |
|---|---|---|---|---|
| Map 1 | 1588 | 1899 | 249 | 75ms |
| Map 2 | 1368 | 1589 | 252 | 70ms |
| Map 3 | 875 | 936 | 90 | 23ms |

The second table summarizes some of the results obtained from Map 1. Data in parentheses indicate when the solution found is an optimal one. To have a better insight of the quality of our solutions we varied the daily load of a mailman, expressed in column 1. The value of $LB$ is given in column 2. Columns 3 and 4 present the results obtained by heuristics $H1$ and $H2$, respectively. Column 5 is split into two columns which contain the results of heuristic $H3$ with $LB$ set to 100 and 500, respectively. Finally, column 6 is also split into two columns which contain the results of heuristic $H4$ with $LB$ set to 100 and 500, respectively.

| Workload | LB | H1 | H2 | H3 | | H4 | |
|---|---|---|---|---|---|---|---|
| | | | | 100 | 500 | 100 | 500 |
| 360 min | 20 | 38 | 23 | 23 | 22 | 23 | 22 |
| 480 min | 16 | 28 | 20 | 18 | (16) | 18 | (16) |
| 600 min | 14 | 24 | 15 | 15 | (14) | (14) | (14) |

Comparing columns 3 and 4, it is clear that the choice of the root node of a cluster in $H2$ largely outperforms the random choice in $H1$. It is interesting to note, however, that the randomized choice, when applied several times, tend to even out its handicap. This can be observed from the results in columns 5 and 6.

The most interesting result is, however, that with the (modified) GRASP algorithms $H3$ and $H4$ we have been able to find the optimal solution for half of the instances we have tested (we can guarantee the optimality of the solutions in parenthesis since they have reached the lower bound). We point out that we have limited ourselves to find solutions which can be computed within one minute, which is quite a severe restriction if we consider that the actual (semi-automatic) postal process for generating a solution takes 120 man-hours.

## 5 Related work

Research on decision support computational environments for urban planning can be divided in two groups: with emphasis on optimization techniques or with emphasis on spatio-temporal concerns. In the first group, results abound in development of heuristics for both graph theoretic and multicriteria algorithms to solve problems in transportation, urban zoning, public utilities allocation. These results, nevertheless, are reported to users by means of graphs or tables (instead of maps), and ignore the advantages of using spatial databases (algorithms more often than not use tables as input).

A frequent compromise is to first solve the algorithmic problem, using non-spatial data, in a given computer environment, and then transport the results to a map visualization tool. Though this solves the visualization problem, it increases data duplication and the probability of introducing errors. An alternative solution is to create a graph from digitized maps, and compute and present the solution on the graph (e.g., like the output of Figure 2). Though this has the advantage of using georeferenced data sources, the output is not adequate.

[NS93] points out the different issues that must be considered in order to use GIS in regional and urban planning, including the demand for optimization techniques. In GIS literature, however, in most cases, urban decision support is based on expert systems and use knowledge bases, and results are obtained exclusively by combining experts' knowledge provided in terms of rules (e.g., [CGND91, LHM94]), without considering the combinatorial aspects of the problem. Examples are found in the context of transportation (e.g., [CAG96]) vehicle navigation and mobile systems (e.g., [Whi91, SY91]) or zoning (e.g., [CGND91]). However, none of these reports discuss the theoretical optimization issues involved.

[HJR94, HJR95] present a recent instance of incorporation of graph theory algorithms into GIS. They combine graph theoretic results with database views and query optimization, in order to optimize computation of paths in transportation systems. The emphasis is on speed of computation rather than optimization of the solution, since some queries have to be answered in real time.

Other examples of combination of GIS and C & O techniques are described in [DBB94, Nie95]. [DBB94] presents an example of the use of GIS in the determination of facility location, incorporating a specific mathematical model to a GIS. [Nie95] describes the use of GIS to support urban transportation and planning. This last paper is in a special issue of a journal which was dedicated to the use of GIS in transportation, but is the only paper to discuss the incorporation of optimization heuristics.

As far as scope and goals are concerned, our work is perhaps closest to that reported in [LRC+92], in the sense that spatial data was processed using an optimization function library, which was developed by C & O specialists. However, [LRC+92] does not seem to have been concerned with issues involved in spatial database design. Furthermore, the applications reported in that work are directed towards routing. Though this is a very important issue, our framework precedes and subsumes a series of combinatorial issues, including routing, since it computes the initial spatial clusters of streets from which mail routing, scheduling and so forth are to be defined. Thus, the nature of the problem is different.

# 6 Conclusions and future work

This paper described part of an ongoing project whose goal is to combine recent results in C & O to spatial databases and GIS technology. The tools and technology developed within this project will support decision taking in the Brazilian Post Office system. We hope that this may help designing systems for a large family of urban decision support applications which require innovative solutions combining optimization, spatial databases and cartographic visualization.

A first prototype has been developed, for decision support in the scheduling of mail distribution districts for the São Paulo Post Office. The prototype takes into consideration not only spatial characteristics but also temporality of mail patterns, and gives users opportunity to produce and compare alternative solutions. The prototype was designed using an open systems paradigm, and is based on coupling an optimization module to a spatial database.

The integration of heuristic algorithms and spatial databases is presenting us with many interesting research and development challenges. From a spatio-temporal data management point of view, this requires research into data modelling and storage structures. From an optimization point of view, changes must be made to allow algorithms to directly manipulate spatial data and support spatial constraints. In fact, just as spatial analysis is not nonspatial analysis on spatial data, heuristics on georeferenced data are not the same as the application of heuristic algorithms on regular data.

This automation liberates planners from time consuming manual tasks, but also brings flexibility from the algorithmic point of view. As well, users are finding out that they can change or design completely new solutions in much less time, reflecting changes in management policy. In addition, since users will be able to electronically manipulate maps, they will be able to perform new kinds of information processing and display which are presently not possible – e.g., [CW94, CWP95].

Besides the obvious time savings due to automation, other main advantages this approach presents in comparison to the current situation are:

- The use of a common database will increase data availability and decrease the amount of planning errors. As well, it will allow faster updates.
- The possibility of changing algorithm input parameters and select heuristics on the fly will help improve existing solutions and compute different types of solutions.
- Since it is based on an open systems philosophy, we are able to extend the prototype progressively and will integrate it with other tools.

As mentioned before, the prototype was used as a means to show Brazilian postal authorities some immediate results. Additionally, important research and development issues are under study. Theoretical research being conducted involves redesign and integration of the present databases and specification of new components of the optimization library. We also intend to extend this project to other scale levels (e.g., distribution zone determination within a city). From an implementation point of view, we aim to allow users to directly interact with the graphical interface in order to query and update the spatial database.

Our first concern was to have a good tool to indicate the minimum number of mailmen needed to ensure distribution and, in this particular, the results show that we have succeeded. We are now working on combining this with the second optimization goal of district determination, namely, workload balance. This is being done by implementing heuristics for the first (local improvement) step of the full GRASP algorithms. The prototype still suffers from the fact that the solutions it generates may have inadequate geometrical district formats from the Post Office point of view. One possible solution is to use a bounding rectangle as an additional restriction in district determination, thereby increasing the spatial constraints imposed on heuristic algorithms.

Another issue will be to connect our interim solution to the interface presentation. This includes solving properly the node coloring problem ([Ber73]) defined on the district graph.

As well, new optimization functions will consider computing workload balance, mailman routing and others.

## Acknowledgements

This work was developed within the PROTEM GEOTEC project and the CEE project ITDC-116, and was partially financed by grants from CNPq and FAPESP.

The authors thank Juan Plaza for his careful review of a preliminary version of this report.

## References

[Ber73] C. Berge. *Graphs and Hypergraphs*. North–Holland, 1973.

[CAG96] C. Caliari, A. Araujo, and F. Gomes. Identificação dos atributos de grafos em imagens de mapas viários para problemas de roteamento.(Identification of graph attributes from the image of road maps for routing problems). In *Proc GIS-Brasil Conference*, pages 388–393, 1996.

[CGND91] J. Chen, P. Gong, R. Newkirk, and G. Davidson. ZoMap: a Knowledge-based Geographical Information System for Zoning. In *Proc GIS/LIS*, 1991.

[CW94] M. Crossland and B. Wynne. Measuring and Testing the Effectiveness of a Spatial Decision Support System. In *Proc. 27th Annual Hawaii International Conference on System Sciences*, pages 542–551, 1994.

[CWP95] M. Crossland, B. Wynne, and W. Perkins. Spatial Decision Support Systems: an Overview of Technology and a Test of Efficacy. *Decision Support Systems*, 14:219–235, 1995.

[DBB94] Y. Ding, A. Baveja, and R. Batta. Implementing Larson and Sadiq's Location Model in a Geographic Information System. *Computers Ops Research*, 21(4):447–454, 1994.

[FK95] A. Frank and W. Kuhn. Specifying Open GIS with Functional Languages. In *Proc SSD'95*, pages 184–195, 1995.

[FR95] T. A. Feo and M. G. C. Resende. Greedy randomized adaptative search procedures. *Journal of Global Optimization*, 1:1–27, 1995.

[GJ79] M. R. Garey and D. S. Johnson. *Computer and intractability: A guide to the theory of NP–completeness*. Freeman, San Francisco, 1979.

[HJR94] Y. Huang, N. Jing, and E. Rundensteiner. Semi-materialized View Approach for Route Guidance in Intelligence Vehicle Highway Systems. In *Proc ACM GIS Workshop*, pages 144–151, 1994.

[HJR95] Y. Huang, N. Jing, and E. Rundensteiner. Hierarchical Path Views: a Model Based on Fragmentation and Transportation Road Types. In *Proc ACM GIS Workshop*, 1995.

[LHM94] P. Longley, G. Higgs, and D. Martin. The Predictive use of GIS to Model Property Valuations. *International Journal of Geographical Information Systems*, 8(2):217–236, 1994.

[LRC+92] G. Lapalme, J-M. Rousseau, S. Chapleau, M. Cormier, P. Cossette, and S. Roy. Georoute: a Geographic Information System for Transportation Applications. *Communications of the ACM*, 35(1):80–88, 1992.

[Nie95] O. Nielsen. Using GIS in Denmark for Traffic Planning and Decision Support. *Journal of Advanced Transportation*, 29(3):335–354, 1995.

[NS93] P. Nijkamp and H. Scholten. Spatial Information Systems: Design, Modelling and Use in Planning. *International Journal of Geographical Information Systems*, 7(1):85–96, 1993.

[PS82] C H. Papadimitriou and K. Stieglitz. *Combinatorial Optimization: Algorithms and Complexity.* Prentice–Hall, 1982.

[SY91] S. Shekhar and T. Yang. Motion in a Geographical Database System . In *Proc. 2nd Symposium Spatial Database Systems*, pages 450–460. Springer Verlag Lecture Notes in Computer Science 525, 1991.

[Whi91] M. White. *Geographical Information Systems - volume II*, chapter Car Navigation Systems, pages 115–125. John Wiley and Sons, 1991.

# Relatórios Técnicos – 1995

95-01 **Paradigmas de algoritmos na solução de problemas de busca multidimensional,** *Pedro J. de Rezende, Renato Fileto*

95-02 **Adaptive enumeration of implicit surfaces with affine arithmetic,** *Luiz Henrique de Figueiredo, Jorge Stolfi*

95-03 **W3 no Ensino de Graduação?,** *Hans Liesenberg*

95-04 **A greedy method for edge-colouring odd maximum degree doubly chordal graphs,** *Celina M. H. de Figueiredo, João Meidanis, Célia Picinin de Mello*

95-05 **Protocols for Maintaining Consistency of Replicated Data,** *Ricardo Anido, N. C. Mendonça*

95-06 **Guaranteeing Full Fault Coverage for UIO-Based Methods,** *Ricardo Anido and Ana Cavalli*

95-07 **Xchart-Based Complex Dialogue Development,** *Fábio Nogueira de Lucena, Hans K.E. Liesenberg*

95-08 **A Direct Manipulation User Interface for Querying Geographic Databases,** *Juliano Lopes de Oliveira, Claudia Bauzer Medeiros*

95-09 **Bases for the Matching Lattice of Matching Covered Graphs,** *Cláudio L. Lucchesi, Marcelo H. Carvalho*

95-10 **A Highly Reconfigurable Neighborhood Image Processor based on Functional Programming,** *Neucimar J. Leite, Marcelo A. de Barros*

95-11 **Processador de Vizinhança para Filtragem Morfológica,** *Ilka Marinho Barros, Roberto de Alencar Lotufo, Neucimar Jerônimo Leite*

95-12 **Modelos Computacionais para Processamento Digital de Imagens em Arquiteturas Paralelas,** *Neucimar Jerônimo Leite*

95-13 **Modelos de Computação Paralela e Projeto de Algoritmos,** *Ronaldo Parente de Menezes e João Carlos Setubal*

95-14 **Vertex Splitting and Tension-Free Layout,** *P. Eades, C. F. X. de Mendonça N.*

95-15 **NP-Hardness Results for Tension-Free Layout,** *C. F. X. de Mendonça N., P. Eades, C. L. Lucchesi, J. Meidanis*

95-16 **Agentes Replicantes e Algoritmos de Eco,** *Marcos J. C. Euzébio*

95-17 **Anais da II Oficina Nacional em Problemas Combinatórios: Teoria, Algoritmos e Aplicações,** *Editores: Marcus Vinicius S. Poggi de Aragão, Cid Carvalho de Souza [Not available.]*

95-18 **Asynchronous Teams: A Multi-Algorithm Approach for Solving Combinatorial Multiobjective Optimization Problems,** *Rosiane de Freitas Rodrigues, Pedro Sérgio de Souza*

95-19 **wxWindows: Uma Introdução,** *Carlos Neves Júnior, Tallys Hoover Yunes, Fábio Nogueira de Lucena, Hans Kurt E. Liesenberg*

95-20 **John von Neumann: Suas Contribuições à Computação,** *Tomasz Kowaltowski*

95-21 **A Linear Time Algorithm for Binary Phylogeny using PQ-Trees,** *J. Meidanis and E. G. Munuera*

95-22 **Text Structure Aiming at Machine Translation,** *Horacio Saggion and Ariadne Carvalho*

95-23 **Cálculo de la Estructura de un Texto en un Sistema de Procesamiento de Lenguaje Natural,** *Horacio Saggion and Ariadne Carvalho*

95-24 **ATIFS: Um Ambiente de Testes baseado em Inje,c ao de Falhas por Software,** *Eliane Martins*

95-25 **Multiware Plataform: Some Issues About the Middleware Layer,** *Edmundo Roberto Mauro Madeira*

95-26 **WorkFlow Systems: a few definitions and a few suggestions,** *Paulo Barthelmess and Jacques Wainer*

95-27 **Workflow Modeling,** *Paulo Barthelmess and Jacques Wainer*

# Relatórios Técnicos – 1996

96-01 **Construção de Interfaces Homem-Computador: Uma Proposta Revisada de Disciplina de Graduação,** *Fábio Nogueira Lucena and Hans K.E. Liesenberg*

96Abs **DCC-IMECC-UNICAMP Technical Reports 1992–1996 Abstracts,** *C. L. Lucchesi and P. J. de Rezende and J.Stolfi*

96-02 **Automatic visualization of two-dimensional cellular complexes,** *Rober Marcone Rosi and Jorge Stolfi*

96-03 **Cartas Náuticas Eletrônicas: Operações e Estruturas de Dados,** *Cleomar M. Marques de Oliveira e Neucimar J. Leite*

96-04 **On the edge-colouring of split graphs,** *Celina M. H. de Figueiredo, João Meidanis and Célia Picinin de Mello*

96-05 **Estudo Comparativo de Métodos para Avaliação de Interfaces Homem-Computador,** *S'ılvio Chan e Heloisa Vieira da Rocha*

96-06 **User Interface Issues in Geographic Information Systems,** *Juliano Lopes de Oliveira and Claudia Bauzer Medeiros*

96-07 **Conjunto fonte máximo em grafos de comparabilidade,** *Marcos Fernando Andrielli e Célia Picinin de Mello*

96-08 **96-08 The Effectiveness of Multi-Level Policing Mechanisms in ATM Traffic Control,** *J.A. Silvester, N. L. S. Fonseca, G. S. Mayor e S. P. S. Sobral*

96-09 **Sequential and Parallel Experimental Results with Bipartite Matching Algorithms,** *João Carlos Setubal*

96-10 **96-10 A CPU for Educational Applications Designed with VHDL and FPGA,** *Nelson V. Augusto, Mario L. Côrtes and Paulo C. Centoducatte*

96-11 **Network Design for the Provision of Distributed Home Theatre Services,** *Nelson L. S. Fonseca, Cristiane M. R. Franco, Frank Schaffa*

96-12 **Modelling the Output Process of an ATM Multiplexer with Correlated Priorities,** *Nelson L. S. Fonseca e John A. silvester*

96-13 **Algoritmos de afinamento tridimensional: exemplos de técnicas de otimização,** *F. N. Bezerra and N. J. Leite*

96-14 **Ensino de Estruturas de Dados e seus Algoritmos através de Implementação com Animações,** *Pedro J. de Rezende e Islene C. Garcia*

96-15 **Sinergia em Desenho de Grafos Usando Springs e Pequenas Heurísticas,** *H. A. D. do Nascimento, C. F. X. de Mendonça N., P. S. de Souza*

*Instituto de Computação*
*Universidade Estadual de Campinas*
*13081-970 – Campinas – SP*
*BRASIL*

reltec@dcc.unicamp.br