

## Software Application for Data Visualization and Interaction in a Location Routing Problem

Andreia Melo, Beatriz Sousa Santos, Carlos Ferreira\*, Joaquim Sousa Pinto

\* Seccção Autónoma de Engenharia e Gestão Industrial, Universidade de Aveiro

**Resumo** – Este artigo descreve uma aplicação<sup>1</sup> para plataformas Windows utilizada na introdução de dados e visualização dos resultados de um problema de localização e distribuição de serviços e materiais obnoxios. O objectivo principal desta aplicação é fornecer, através de uma interface “usável”, uma forma fácil e intuitiva de introduzir e visualizar a informação necessária para a resolução do problema e de visualizar e analisar a correspondente aos resultados. Descrevem-se as técnicas utilizadas no desenho do diálogo e na visualização da informação.

**Abstract** - This paper describes an application developed for Windows platforms, which is meant to be used for inputting data and visualizing results of a Location-Routing problem of obnoxious services and materials. The main goal of this application is to provide, through a “usable” user interface, an intuitive of inputting the information needed to solve the problem as well as to visualize and analyze the results. This paper also describes the techniques used both for designing the dialog and for information visualization.

### I. INTRODUCTION

The problem of locating obnoxious facilities (i.e. with undesirable effects, such as warehouses containing flammable materials, regions containing waste materials, garbage dumps, sewage plants) has become a major social concern. In these cases the traditional optimality criterion of “closeness”, (i.e. a desire to locate the facility as close as possible to the customer) is replaced by the opposite criterion, i.e., how far away from the customers can the facility be placed ensuring accessibility to the demand points, or the NIMBY syndrome (Not-In-My-Back-Yard). In these problems location and routing decisions are intimately related and combining both provides, in fact, a better modeling. However, their combined use generally increases computational complexity; moreover this complexity is also increased due to the fact that using distance as the only criterion is, in most real cases, a “poor” approximation since there are other (generally conflicting) criteria of interest to consider simultaneously (as demand, different types of cost, risk, equity,

environmental issues, etc.). Several multiobjective models for the location-routing of undesirable facilities consider that the set of acceptable locations was previously identified which, in fact, corresponds to a common practice in real situations. Although these methods could be used without the support of specifically developed computational means, their application to real-life cases will, in most circumstances, be jeopardized by the huge demanded effort. As a consequence the use of graphical and interactive methods becomes of the utmost importance [1].

In this paper we describe a software application developed for Windows platforms, in Visual Basic 5.0™, which is meant to provide a user interface for a Location-Routing problem in the area of obnoxious facilities.

The developed application is concerned with solving the problem of collecting wastes from communities to be disposed or processed in more than one type of facility. This problem is solved using a multiobjective linear integer programming model (which includes not only the determination of facility locations but also the routing i.e., routes as well as number, type and capacity of vehicles) and is far more complex as to the information to visualize and interact with than other models used previously [2; 1].

The main objective of this application is to offer a “usable” user interface, providing an intuitive way of inputting and visualizing all the information needed to solve the specific Location Routing problem, as well as visualizing the information corresponding to the several alternative solutions that can be obtained. This information is in large quantity and of different nature making its visualization a difficult task.

In the following sections a general overview of the application is given as well as some details of its more relevant features.

### II. OVERVIEW OF THE APPLICATION

The application is concerned, as referred, with the problem of collecting wastes from communities to be disposed or processed in more than one type of facility (service). These communities are located at some nodes of a network and the wastes must be transported along the links of that network to other nodes (where the services are installed). The Multiobjective Location Routing model used is a new bicriteria model [3] formulated as a Mixed Integer Linear Programming Problem. In this model one

<sup>1</sup> O estudo do problema de visualização da informação e a implementação da aplicação foram feitos, respectivamente, no âmbito das disciplinas de Visualização por Computador e Interfaces Humano Computador do Mestrado em Eng. Electrónica e de Telecomunicações em 1997/98.

objective function minimizes costs (travel cost for the population/community, investment as well as operating costs for the installed services), whereas the other objective function minimizes obnoxious effect from the installed facilities as well as the risk associated to the routes used by the vehicles. The set of constraints ensures that all the communities are served, each by only one vehicle, each vehicle operates at most to one service, the maximum capacity of each service is not exceeded and no more than one service of each type is installed on each potential location. This means that the following data needs to be input to the system:

- location of each community to be served
- location of each potential service site
- available vehicles to transport the materials from the communities to the services
- total demand of each community
- different types of vehicles (meaning different capacities)
- distances (shortest path over the network links and Euclidean) among communities and services
- risks associated with each link
- investment and operating costs of the services
- obnoxious effect of each type of service on the communities.

Using this information the efficient solutions (alternatives) of the problem can be determined interactively by the user (through optimization software[4]) who will decide which to choose based on the characteristics of each alternative displayed graphically. Each alternative generates the following output data: a set of installed services (on some of the potential locations), an assignment of communities to services, the number and type of vehicles to be used and their routes.

The introduction of data, its visualization and that of the output data involves several interaction and visualization issues, such as:

- general organization of the interface
- input data techniques (e.g. numerical versus graphical)
- graphical representation of data concerning input or solution characteristics (e.g. visual code used to represent different values of each type of data as cost or risk)
- using time to facilitate the perception of the information (e.g. animation of vehicle routes of each solution or visualization of input data or solution characteristics at different times)
- calibration techniques for input distances

These and other issues were studied and solutions for our particular case were developed based on the principles and guidelines in the literature of Human Computer Interaction and Visualization [5; 6; 7; 8; 9; 10; 11, 12].

### III. USER INTERFACE

The user interface is organized around two fundamental windows, the Main Window and the Maps Window, shown respectively in Figs. 1 and 2. The Main Window is used to access all the functionality of the application and to input the most important data; through this window it is possible to specify the number of communities, services,

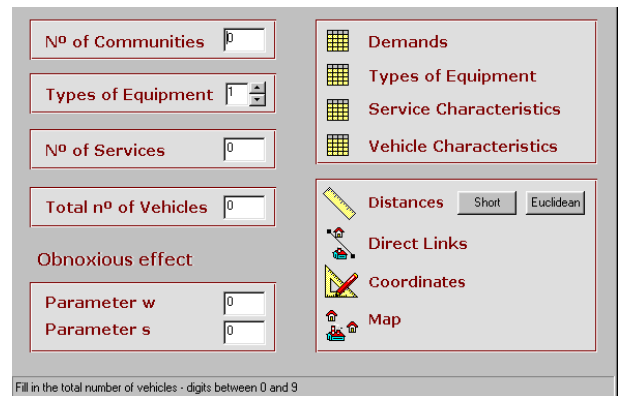


Fig. 1 – Main Window

vehicles, types of services, the parameters corresponding to the obnoxious effect (W and S) and open other windows to input some more input related data (such as communities and potential service locations, vehicle characteristics and distances among communities and services).

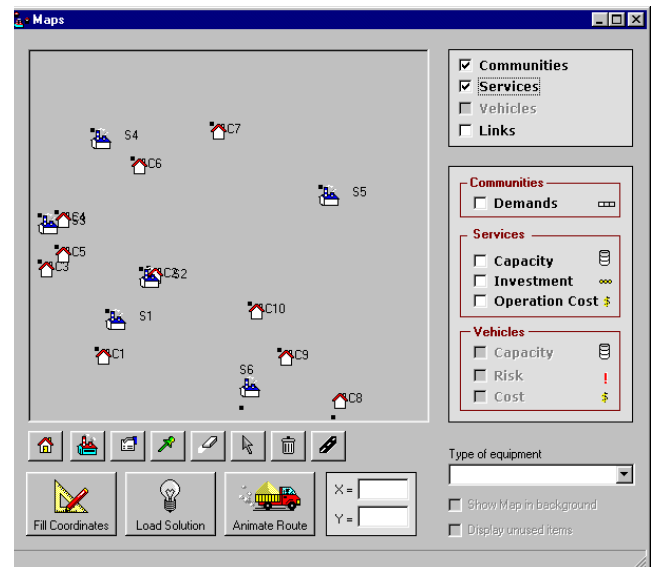


Fig. 2 – Maps window (displaying only input communities and potential location of services, without any map)

The Maps Window allows the graphical input of some data and the visualization of input and output information. This window has several features, including:

- a working area, where it is possible to input graphically the location of communities and services

- and the links among these locations as well as to visualize other information
- the possibility of loading a map (in BMP, GIF or JPEG) which gives context to the user on a real situation, making easier to input and understand the information
- a set of graphical primitives and actions to use on the working area available through a toolbar (bellow the working area)
- a set of controls (on the right) which allows displaying or hiding information on the working area; each control can be used to visualize input or output information
- the possibility to get efficient solutions of the problem (through the button “Load Solution” that call an optimization S/W to obtain what we have been calling the output data)
- the possibility of animating the vehicles routes corresponding to each efficient solution (through the button “Animate”).

When the Maps Window is maximized the area occupied by the working area (and consequently by the map) increases to its maximum, thus allowing a better visualization of graphical data. Since the map scale varies the application provides two means for specifying it: through the specification of a known distance on the map or entering the coordinates corresponding to the maximum width and maximum height of the map. These two methods are exclusive and can be accessed through the application menu. The first is based on an assistant, which gives instructions to the user.

#### IV. VISUALIZATION OF THE INFORMATION

Each efficient solution of the problem generates the following output data: a set of installed services (on some of the potential locations), an assignment of communities to services, the number of vehicles to be used and their routes.

All the information concerned with the problem definition and its efficient solutions (input or output data) needs to be available to the user. This is done through tables and graphically. The following information is visualized on the working area of the Maps Window:

- background map – problem context
- communities
- demands of service of all communities
- services
- capacity of each service
- necessary investment for each service
- cost of operation of each service
- type of service
- vehicles
- capacities of each vehicle
- risk associated to each vehicle
- cost of each vehicle

- all links among communities and services
- routes followed by each vehicle
- all routes simultaneously
- single route activation and indexes of communities(C1, C2,...) and services (S1, S2,...).

The visualization of the locations (communities or services) makes the perception of the distances implicit from the map. All the information is visualized layer by layer on the working area activating and deactivating controls (check boxes). This approach provides a great flexibility to the user since it is very easy to select the set of information useful at any time.

#### A. Graphical representations

A graphical representation (icon) is assigned to each type of entity and related characteristics that have to be visualized on the visualization area (communities, services, vehicles, links and routes).

The icons used to represent communities, services and vehicles depict objects easily associated with them such as houses, factories and trucks, respectively. The use of these icons is expected to result intuitive to the user. Each of these icons has a black dot and an index used to mark its location and indicate the index of the corresponding entity. These marks are used as a minimal representation.

Lines are used to represent links; this is a rough approximation to the real situation, however the important information is the existence of links and this method is very simple and seems to result not too confusing.

Vehicle routes link communities and services in a closed circuit. The simultaneous visualization of the routes corresponding to an alternative may result very confusing, thus animation of single routes is used, as described in more detail in section IV.C.

When much information has to be displayed simultaneously, icons can become overlapped in such a way that sometimes they look like a single icon; this has to be taken into consideration when designing the icon.

Fig. 3 shows an example of the service characteristics that are represented by four individual icons:

- a factory, representing the service;
- a 2/3 full recipient, representing a medium service capacity;
- one coin, representing a low service investment
- three dollars; representing a high cost of operation.



Fig. 3 – Graphical representation of all service characteristics.

There are two types of graphical representation of communities and services since the input and output data have to be visualized in different ways. The first case (input) corresponds to the problem definition so we use draft symbols. In the second case (output), which corresponds to the solution, we have a more detailed

representation. All symbols for both cases are represented in Fig. 4. This differentiation is based on the fact that some services are not implemented by the solution and it is useful to represent simultaneously both types of services (those not implemented by the solution are represented by the draft symbol equal to the one used for data input).

In order to avoid ambiguities, the visualization of communities, services and vehicles characteristics is made only if their symbols are present on the map. This constraint becomes necessary since the symbols used to represent the capacity and cost are the same for services and vehicles (as can be noticed in Fig. 4).

The use of graphical symbols to represent the characteristics (which are in fact variables that can assume values in an interval) allows only the representation of a small number of different values, for the sake of clarity. Thus, to avoid confusion and difficult interpretation of the map, we have decided to represent only three levels of characteristics: low, medium and high. The numeric values can be assessed from the tables.













	Data Input	Results Visualization
Communities		
Characteristics	Demand of service: 	
Services		
Characteristics	Capacity:  Investment:  Operation Cost: 	
Vehicles		
Characteristics	Capacity:  Risk:  Cost: 	

Fig. 4 – Graphical representation of communities, services, vehicles and their characteristics.

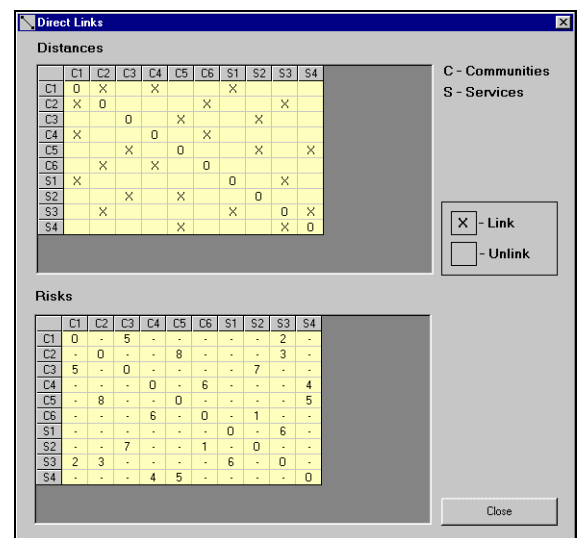
The icons representing graphically the communities, services, vehicles and their characteristics, are shown in Fig.4. Every icon used to represent a characteristic is either divided in three parts (Demands and Capacities) or composed by up to three small equal symbols. In the first case, Demands and Capacities are represented by a horizontal bar and a stack, which can be filled in three different ways, corresponding to three levels (low, medium, high). In the second case, Investment, Operation Cost and Risk are represented by one, two or three equal

symbols (coins, exclamation marks and dollars) according to the value (also three levels). In order to choose how many symbols will be used to represent a specific value of a characteristic, two adequate intermediate levels (from the range of possible values) are used.

*B. Defining direct links*

Direct links among communities and services are established by the user either using a table or graphically on the map. The table is represented in Fig. 5, it offers a very efficient way of creating links between any two locations (communities and services) and it is the only method that allows erasing a link. In any case, it is possible to use both methods together. To create a link on the table, the user must select the corresponding cell and a symbol “X” will be automatically entered, indicating that the community/service of that row is connected to the community/service of the corresponding column.

A risk must be assigned to each link, which can be done using another table (as shown also in Fig. 5).



	C1	C2	C3	C4	C5	C6	S1	S2	S3	S4
C1	0	X		X			X			
C2	X	0				X			X	
C3			0		X			X		
C4	X			0		X				
C5			X		0			X		X
C6		X		X		0				
S1	X						0		X	
S2		X		X				0		
S3			X				X		0	X
S4					X				X	0

	C1	C2	C3	C4	C5	C6	S1	S2	S3	S4
C1	0	-	5	-	-	-	-	2	-	-
C2	0	-	-	8	-	-	-	3	-	-
C3	5	-	0	-	-	-	-	7	-	-
C4	-	-	0	-	6	-	-	-	4	-
C5	-	8	-	0	-	0	-	-	-	5
C6	-	-	-	6	-	0	-	1	-	-
S1	-	-	-	-	-	-	0	-	6	-
S2	-	-	7	-	-	-	-	0	-	-
S3	2	3	-	-	-	-	6	-	0	-
S4	-	-	-	4	5	-	-	-	-	0

Fig. 5 – Table of Direct Links among communities and services.

To establish a link graphically it is necessary to select the source and the destination locations (community or service) on the map. The source is marked with a green pin and a rubber band echo [13] is used to help the user defining the right link. If the user selects a second time the same location it will be unmarked and it will be possible to choose another source. This procedure is illustrated in Fig. 6 and the visualization of all links is shown in Fig. 7.

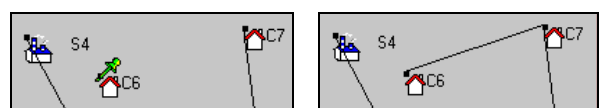


Fig. 6 – Establishing a link graphically: on the left, C6 was marked as the source point and on the right C7 was the destination point.

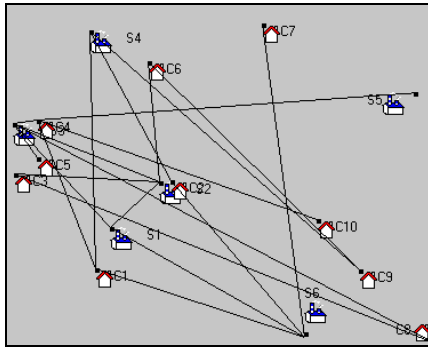


Fig. 7 – Representation of all links in the map

C. Animation of routes

A route is represented graphically as a set of links forming a closed circuit. It begins on a service, passes through several communities and ends on the same service. The information relative to these routes is displayed on the Routes Window shown in Fig. 8. It contains a list of all the routes of each alternative in textual form. For instance, S3 – C2 – C1 – S3 means that a vehicle leaves service S3, passes the community C2 and after C1 and returns to S3. This information can not be modified (as it is output data).

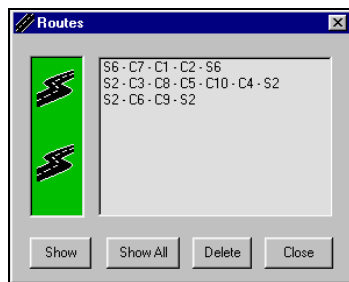


Fig. 8 – Routes Window

An effective visualization of the routes is not easy to obtain since the visualization area can become too cluttered. As animation is a very efficient form to visualize a continuous phenomenon [8], it was used to visualize a route at a time (as referred in IV). Five images are displayed sequentially (Fig. 9), giving the illusion of a vehicle moving along the route.



Fig. 9 – Bitmap images used for animation

The first four bitmaps are almost equal, except for the background color (black and white). A white bitmap is necessary to provide the background transparency when the vehicle is moving (a BitBlt function and a XOR [13]

are used to produce bitmaps containing the vehicle and the right background).

A timer was used to determine the frequency used to show the bitmaps (1/150ms).

When the animation is started it is necessary to compute the slope of the line corresponding to the first link. This slope is used to determine intermediate coordinates where the vehicle bitmap will be placed in order to move smoothly along the line. This is repeated each time the vehicle reaches a community and has to start following another link (having a different slope). When a vehicle is moving its route is activated (i.e. its color changes).

D. Changes in graphical representation

The visualization of communities and services (using the icons of Fig. 4) could be difficult for the ones located very near the boundaries of the visualization area. In these cases we must use different icons to represent them, to ensure that the entire icon is visible and understandable. Fig. 10 contains all icons used to represent communities and services according their relative location. The black squares and the indexes indicate the entity location and they may vary depending on the used scale. The symbols and indexes Cx and Sx are always placed as to guarantee their complete visibility.

The default symbol used to represent a community or service is equal to the one used for the upper left corner. The choice of the symbol is made at runtime. This procedure can be observed by moving the position of an icon. The user may drag and drop the icon and it will automatically be adjusted to the new location. If the distance from the location to the boundary is less than the default icon width or height another icon must be used.

	Upper left corner	Upper right corner	Bottom right corner	Bottom left corner
Community				
Community in solution				
Service				
Service in solution				

Fig. 10 – Graphical representation of communities and services depending on their map coordinates.

The symbol used to mark the location of a direct link source (the green pin) point always to the black dot (of the community or service) and depends also on the distance to the boundaries. All cases are presented in Fig. 11. The same is valid for the services.





If the coordinates are too close of the map		
Left boundary	 (by default)	
Right boundary		

Fig. 11 – Graphical representation of communities depending on their location.

### E. Selection

Single and multiple selection of communities and services are allowed. Multiple selection may be interesting, namely to delete several entities simultaneously. Vehicles can also be selected but only to indicate which vehicle will be used for animation. All cases of selection are represented in Fig. 12.



Fig. 12 – Selection of communities, services and vehicles.

### F. Moving icons

Moving a community and/or a service in the visualization area corresponds to modifying its coordinates. This can be achieved by dragging the desired item using the mouse. While a symbol is dragged the source coordinates aren't updated until the mouse button is released on the new destination. When the mouse movement starts, the original symbol remains in the source location and a copy is dragged near the mouse cursor with a yellow background (as shown in fig. 13).

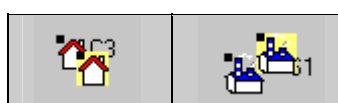


Fig. 13 – Dragging communities and services in the map.

When the mouse button is released, the old icon disappears and a new one (chosen as indicated in fig. 10) is placed on the new location.

## V. CONCLUSIONS

A prototype of an application for information visualization and user interaction in the area of obnoxious facilities was presented. This application is concerned with the problem of collecting wastes on communities to be disposed or processed in several types of facilities. Although this application corresponds to a particular Location Routing problem we believe the addressed issues and many of the solutions proposed are relevant to other, less specific, decision making scenarios where large quantities of information as well as usable interaction procedures have to be presented to the user.

## IV. REFERENCES

- [1] Ferreira C., Sousa Santos B., Captivo M.E., Clímaco J., Silva C.C., "Multiobjective Location of Unwelcome or Central Facilities Involving Environmental Aspects – a Prototype of a Decision Support System", *Belgian Journal of Operation Research, Statistics and Computer Science* 36 (1996) 159-172
- [2] Pinto, S., Sousa Santos B., Ferreira C., Rafael J., "Sistema de Apoio à Decisão para Problemas de Cobertura: definição e implementação de uma interface de utilizador", *Revista do DETUA, Vol.1, N.4*, 1995, 363-366
- [3] Ferreira C., *Problemas de Localização e Distribuição Multicritério – Aproximações e estudo de alguns casos com Implicações Ambientais*, dissertação de Doutoramento, Universidade de Aveiro, 1997.
- [4] CPLEX Optimization, Inc., *Using the CPLEX Callable Library (v. 3.0)*, Incline Village, Nevada, USA, 1994
- [5] Mayhew, D., *Principles and Guidelines in Software User Interface Design*, Prentice Hall, 1992
- [6] Dix, A., J. Finley, G. Abowd, B. Russell, *Human Computer Interaction*, Prentice Hall, 1993
- [7] Shneiderman, B., *Designing the User Interface, Strategies for Effective Human-Computer Interaction*, 3rd ed., Addison Wesley, 1998
- [8] Brodlie, K., L. Carpenter, R. Earnshaw, J. Gallop, R. Hubbard, A. Mumford, C. Osland, P. Quarendon, *Scientific Visualization, Techniques and Applications*, Springer Verlag, 1992
- [9] Keller, P., M. Keller, *Visual Cues*, IEEE Computer Society Press, 1993
- [10] Brown, J., R. Earnshaw, M. Jern, J. Vince, *Visualization, Using Computer Graphics to Explore Data and Present Information*, John Wiley, 1995
- [11] Tufte, E., *The Visual Display of Quantitative Information*, Graphics Press, 1983
- [12] Nielsen, J., *Usability Engineering*, Academic Press, 1993
- [13] Foley, J., A. van Dam, S. Feiner, J. Hughes, *Computer Graphics: Principles and Practice*, 2nd ed., 1990