Calibrating Model Based On Anticipation By Genetic Algorithms

Application To Real-Time Demand Responsive Transport

Jerome Bolot, PhD student
Thema, CNRS UMR 6049, Besançon, France
Jerome.bolot@univ-fcomte.fr

Abstract. This paper concerns real-time Demand Responsive Transport. From the user's point of view, it implies booking, optimising both schedules and routes in real time. Nevertheless, this optimisation isn't free of constraints, as we must take into account the possible booking of other users, as well as the main characteristics of the vehicle fleet (number and type of vehicles). The main point of our research is to propose a model based on anticipation to approach the potential number of users in each station to be served. This potential is based on the past registered reservations. To calibrate the model parameters, we propose to use genetic algorithms, potentially very efficient with this kind problem. In the meantime, each model built by the generator is evaluated and when the process stops, the system keeps the best solution among the whole set.

1. INTRODUCTION

A French Demand Responsive Transport (DRT) called Evolis-gare transports TGV's users to the train station (figure 1). It's necessary for user to book by phone before the day of his transport. This operation permits to optimize path and use the right number of vehicles by increasing the number of client in each vehicle. This service is in order since october 2000 by the collaboration between a public carrier and the laboratory Thema-CNRS.

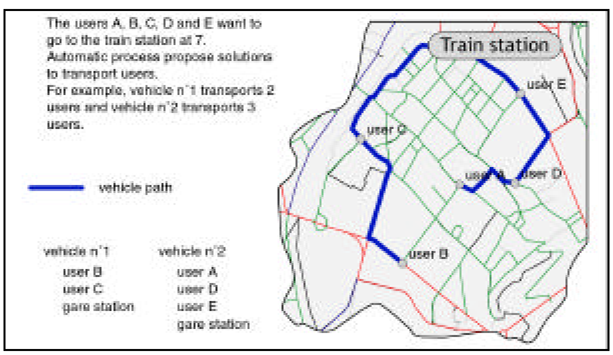

Figure 1: Evolis-Gare, a French DRT

We use this experience to build our research. Indeed, we want to propose a similar service with a real-time booking. This system means that companies in charge of RDT know the real demand only when it occurs. The consequence is the impossibility for public carrier to prepare the necessary fleet of vehicle early. This is an important problem. We need to anticipate the users calling to locate the right number of vehicles at the right locations.

The process we intend to set is quit classical. First we use some adequate data to model the studed geographical space then, among this available data, we identifie those wich may be the releable indicators. Finally, this indicator are computed within specific parameters used by optimisation methods. In other terms, the transport demand is describe by various information (station locations, network, client). This information can be modeled, aggregated, computed to provide specific indicators (example diection possible).

We know genetic algorithms are very efficient to resolve problems.

Thus, we'll present in the paper two different aspects :

- The tool we develop for modeling and visualizing the spatio-temporal information in order to makes series of hypotheses generated by spatio-temporal exploratory,
- A genetic algorithm is used to combine these hypotheses to find the right model parameters.

2. THE HIGHLY SPATIO-TEMPORAL INFORMATION

Since october 2000 the DRT Evolis is working . From this date we store the booking and the routes made by the vehicles. So, can we observe how evolves the demand in time each day. To assess the potential demand at any time, we must observe the characteristics of moments and places. That's why we develop a specifical computational environment.

Let see now what are the data available and the dedicated visualization tools (time and location views).

2.1 Data available

The data used have temporal and spatial characteristics (figure 2).

- **User information** concerns the DRT client. Identification of each of them permits to observe frequencies,
- **Station** is used to locate the position where vehicles take user,
- **Calendar** informs about the particularity of each day (holiday, public holiday),
- **Booking** is the link with all information.

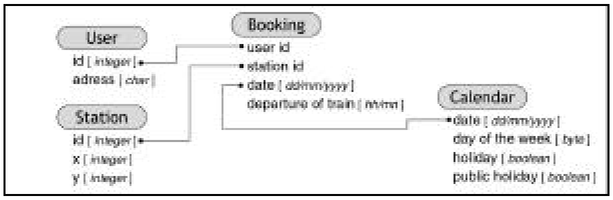

Figure 2: Data available

Spatial appears with the station whereas temporal effects are controlled by a calendar. The information is not directly implemented within a GIS. In fact, we need speed computing and not necessary all classical functions available on GIS. That's why a special interface, TDI 2 developed in Delphi language permits to access the data easily in total respect to the model.

2.2 Visualization

Our tool provides to locate the clients on a network background map. We use MapInfo format (MIF) for graphics data. The relative coordinates permit some classical visualization's operations like zoom or screen moving. Windows DirectX 8 library are used to accelerate the display.

More over a specific window we develop, we can select time events and intervals. This window is directly links to the graphic map. The demand evolution in space and time becomes then easy to explore (Tukey).

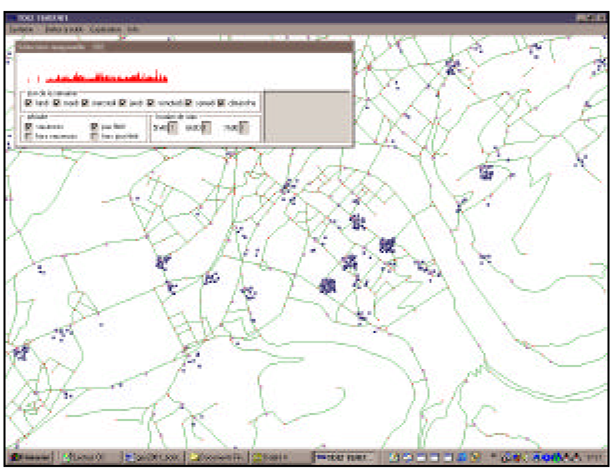

Figure 3: Visualization of the information

2.3 Some questions about spatio-temporal information on RDT

Is there a relation between each same day of the week ? Are the weeks alike ? Are the week-end alike too ? Are the week during holidays different than normal weeks ? Is the spatial or temporal repartition chaotic ?

For instance, we think the service isn't enough important to use statistic tests to verify all these hypotheses. However, a first Poisson's test (figure 4) based on the total space permits to see a logic about spatial distribution but only for the small number of the distribution.

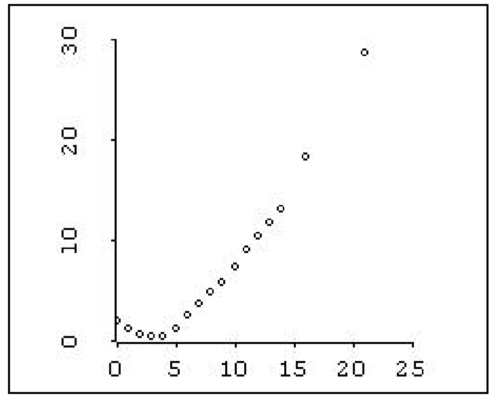

Figure 4: Poissonness plot

Two reasons explain this a priori random distribution :

- a particular local behavior. The demand of RDT hasn't an equal repartition all around the space. In fact, this particular service is rather attractive for a particular population.
- The service is quite new and is no already stable.

These reasons require to propose a model with local parameters. These parameters must be reported to a partitioned space.

2.4 An appropriate partionning ?

The partition is built by a grid. The figure 5 presents two possible forms of the cell :

- A square cell (see on left),
- An hexagonol cell (see on right).

A part of our further work will be to test the accuarancy of each kind of partions.

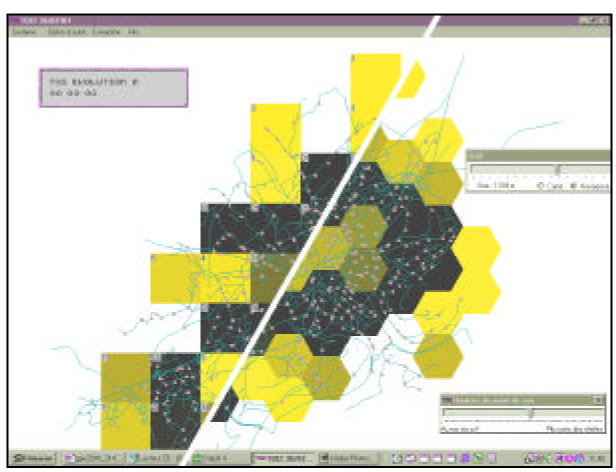

Figure 5: Two ways for spatial partitionning

At this time, as we saw booking spatial density and temporal distribution (see 2.2) are the relevant local indicators. Those are graphics and help user to define grid size and shape. Later, we'll search an automatic indicator allowing to find the appropriate size and form.

2.5 The parameters of the potential demand assessment

Different points of this chapter can constituted model parameter : repetition of relation between days or weeks, size of booking sample and spatial resolution. This last point won't be defined by the model (figure 6). In fact, we think that integration of this point in the model complex it.

Nevertheless, we propose to distinguish the near past of far past by a coefficient (b in figure 6).

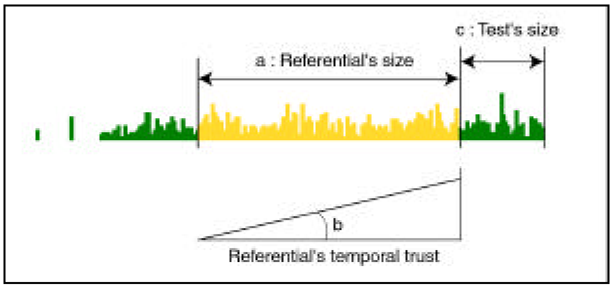

Figure 6: The model's parameters

However, information is now ready to be computed by genetic algorithms.

3. GENETIC ALGORITHMS

When complexity is high and when heuristic methods don't lead to sufficient result, we can try genetic. Using genetic algorithm to compute geographical information seems to be a promising way (Contassot-Vivier and Chatonnay, 1999), and it exists other interesting experiences (Cohoon, 1986, Diplock, 1996, Turton and al., 1997)

The conception of a genetic algorithm is generally decomposed in four main parts (for further information see Goldberg 1989) :

- Creation of the initial population and element representation,
- Evaluation of each element in the population,
- Computation of the new population (crossing-over and mutation),
- Choice of the convergence condition.

In this chapter we discuss each of these four points according to the problem we want to address

3.1 Creation of the initial population

First of all, we have to describe our initial data in a way that allows us to manipulate them as genetic information.

The size of an individual is equal to the number of the model necessary parameters (see 2.3 and 2.5). Here is an example of a parameters vector on the whole set of client (figure 7)

Figure 7: Parameters vector coding

The initial population is build by a random process. We obtain a set of individuals that makes our population. The number of initial points is chosen according to the size of the population we want. By this way, we ensure that the initial population is diversified enough to allow a good convergence. Once the population is created, we have to evaluate each individual.

3.2 Individuals evaluation

The evaluation must give a score for each individual, which directly depends on the accuracy of the anticipation deduced from this individual. To calculate the score, we compare the model defined by referential to the test area (figure 6).

Also, according to the algorithms we use, the set of scores must be normalized and processed to have the properties of a probabilistic law. Hence the score of an element represents the probability to stay or to be a parent in the next population..

3.3 Crossing-over and mutation

Once the elements of the current population are evaluated, we have to construct the new population. To do so, we randomly select couples of parents (figure 8).

individuals	score	part of score
A	37	5 %
B	44	6 %
C	52	7 %
D	103	14 %
E	111	15 %
F	170	23 %
G	222	30 %
total	740	100 %

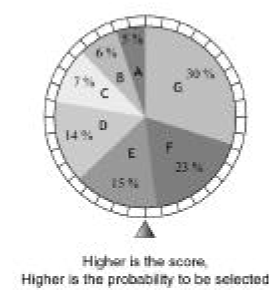

Figure 8: The individual's random selection

The aim of this type of selection is to delete rather the bad elements. Favoring the selection of good elements permits to improve the population. In fact, the parents are used to build two children by crossing-over and mutation (figure 9-a).

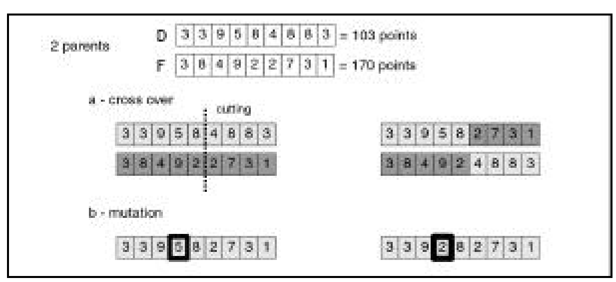

Figure 9: Two parents generate two children

To make the crossing-over, the matrix representing the two elements is randomly divided in two parts and the parts are exchanged. The second part of the second element is merged to the first part of the first element and the first part of the second element is merged with the second part of the first element.

Moreover, we ensure a good variation of the population by introducing mutation (figure 9-b). Once two elements are created using the crossing-over method, we randomly decide to apply or not a mutation. When mutation is applied, one of the two children is randomly chosen to mute.

3.4 First results

We did a few tests by implementing ga. We first had to define a convergence criterium. We choose to stop genetic algorithms after 1000 iterations. The figure 10 focusses on a step of the convergence process. For instance, the geographical space is devided into equal square and the black cell (buttom) is processed at this time

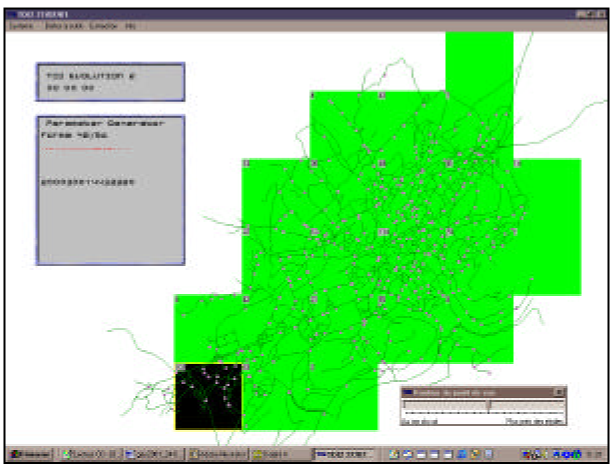

Figure 10: Parameter's generator

In this application, each cell has each own population of parameters and convergence of genetic algorithms is showed in the figure 11. The progression of result quality is gradual.

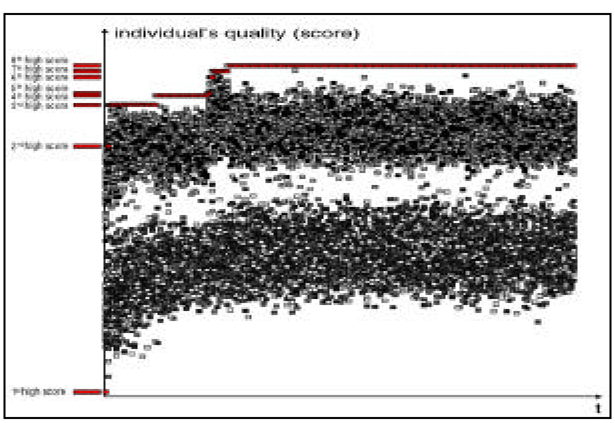

Figure 11: Individual's evaluation

Thus, we obtain good results with high-density information cell (where lots of clients are located, see 2.4). But, it appears hazardous to anticipate something on the low-density cell because they are unforeseeable.

4. CONCLUSION

We can resume the approach by three points.

At first, we define the partitioning of space with exploratory data. This method permits to reduce the complexity model. We will try to integrate this parameter later.

Genetic algorithms permit a large automatic exploration of solutions. We expect integrate more parameters

In point of view of application, cells with important demand can be anticipated. In fact, for public carrier compagnies, the important point is to estimate the main part of demand. Real-time systems will provide more efficient modeling of demand change integration.

REFERENCES

Bolot, J., Chatonnay, P., Josselin, D. (1999), "Construction and evaluation of spatial partitions to describe geographical flows", International Symposium on Spatial Data Quality 99, Hong-Kong, July 1999, pp 523-533

Bosq D., Lecoutre, J.P. (1992), Analyse et prevision des series chronologiques, Methodes parametriques et non parametriques, Masson, 203 p.

Box, G., Jenkins G. (1970), Time series analysis, forecasting and control, Holden Day

Cohoon, J.P., Paris, W.D. (1986), "Genetic Placement", Proceedings of the 23rd, Design Automation Conference, pp 422-425

Contassot-Vivier, S., Chatonnay, P. (1999), A genetic algorithm for high accuracy in computation of epipolar geometry of two uncalibrated views, Research report of Laboratoire d'informatique de Franche-Comte, 5 p.

Goldberg, D. E. (1989) Genetic Algorithms in Search, Optimization, and Machine Learning, Addison-Wesley, Reading, Massachusetts, 412 p.

Tukey, J. (1977) Exploratory data Analysis, Addison-Wesley, Reading, Massachusetts, 688 p.

Turton, I., Openshaw, S. and Diplock, G . (1996) "Some geographical applications of genetic programming on the Cray T3D. in C", Jesshope and A. Shafarenko (eds). UK Parallel'96, Springer, Berlin, pp 135-150