# An Interactive Train Scheduling Tool for Solving and Plotting Running Maps*

F. Barber[1], M.A. Salido[2], L. Ingolotti[1], M. Abril[1], A. Lova[3], P. Tormos[3]

[1] DSIC, [3] DEIOAC, Universidad Politécnica de Valencia, Spain
[2] DCCIA, Universidad de Alicante, Spain
{fbarber, msalido, lingolotti, mabril }@dsic.upv.es
{allova, ptormos}@eio.upv.es

**Abstract.** We present a tool for solving and plotting train schedules which has been developed in collaboration with the National Network of Spanish Railways (RENFE). This tool transforms railway problems into formal mathematical models that can be solved and then plots the best possible solution available. Due to the complexity of problems of this kind, the use of preprocessing steps and heuristics become necessary. The results are plotted and interactively filtered by the human user.

## 1 Introduction

Over the last few years, railway traffic have increased considerably, which has created the need to optimize the use of railway infrastructures. This is, however, a hard and difficult task. Thanks to developments in computer science and advances in the fields of optimization and intelligent resource management, railway managers can optimize the use of available infrastructures and obtain useful conclusions about their topology.

The overall goal of a long-term collaboration between our group at the Polytechnic University of Valencia (UPV) and the National Network of Spanish Railways (RENFE) is to offer assistance to help in the planning of train scheduling, to obtain conclusions about the maximum capacity of the network, to identify bottlenecks, to determine the consequences of changes, to provide support in the resolution of incidents, to provide alternative planning and real traffic control, etc. Besides of mathematical processes, a high level of interaction with railway experts is required to be able to take advantage of their experience.

Different models and mathematical formulations for train scheduling have been created by researchers [9], [3], [4], [8], [6], [2], [5], [1], [10], etc. Several European companies are also working on similar systems. These systems include complex stations, rescheduling due to incidents, rail network capacities, etc. These are complex problems for which work in network topology and heuristic-dependent models can offer adequate solutions.

---

* This work has been supported by a join contract RENFE-UC/UPV and a visiting research fellow (Miguel A. Salido) from the Polytechnic University of Valencia, Spain.

In this paper, we describe a specific tool for solving and plotting optimized railway running maps. A running map contains information regarding the topology of the railways (stations, tracks, distances between stations, traffic control features, etc.) and the schedules of the trains that use this topology (arrival and departure times of trains at each station, frequency, stops, junctions, crossing, overtaking, etc,)(Figure 1). An optimized running map should combine user requirements with the existing constraints (railway infrastructures, user requirements and rules for traffic coordination, etc.). In our system, the railway running map problem is formulated as a Constraint Satisfaction Problem (CSP) to be optimized. Variables are frequencies, arrival and departure times of strains at stations. The parameters of the process are defined using user interfaces and database accesses. The problem formulation is then translated into a formal mathematical model to be solved for optimality by means of mixed integer programming techniques. Due to the dimensions of the problem and the complex nature of the mathematical models, several preprocesses and heuristic criteria are included in order to obtain good solutions in a reasonable time. The user can also modify the obtained timetable so that the system interactively guarantees the fulfillment of problem constraints by detecting whether a constraint or requirement is violated. Several reports can be obtained from the final timetable.

## 2 Problem Statement

A sample of a running map is shown in Figure 1, where several train crossings can be observed. On the left side of Figure 1, the names of the stations are presented and the vertical line represents the number of tracks between stations (one-way or two-way). The objective of the system is to obtain a correct and optimized running map taking into account: (i) the railway infrastructure topology, (ii) user requirements (parameters of trains to be scheduled), (iii) traffic rules, (iv) previously scheduled traffic on the same railway network, and (v) criteria for optimization.

A railway network is basically composed of stations and one-way or two-way tracks. A dependency can be:

- **Station**: Place for trains to park, stop or pass through. Each station is associated with a unique station identifier. There are two or more tracks in a station where crossings or overtaking can be performed.
- **Halt**: Place for trains to stop, pass through, but not park. Each halt is associated with a unique halt identifier.
- **Junction**: Place where two different tracks fork. There is no stop time.

In Figure 1, horizontal dotted lines represent halts or junctions, while continuous lines represent stations. On a rail network, the user needs to schedule the paths of n trains going in one direction and m trains going in the opposite direction, trains of a given type and at a desired scheduling frequency.

The type of trains to be scheduled determines the time assigned for travel between two locations on the path. The path selected by the user for a train trip

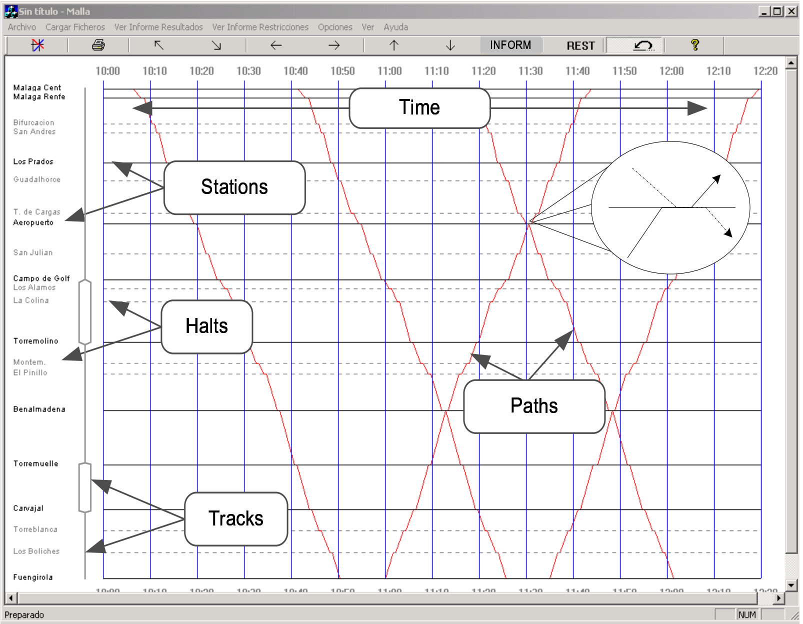


**Fig. 1.** A sample of a running map

determines which stations are used and the stop time required at each station for commercial purposes. New trains to be scheduled must be coordinated with previously scheduled trains. In order to perform crossing or overtaking in a section with a one-way track, one of the trains should wait in a station. This is called a *technical stop*. One of the trains is detoured from the main track so that the other train can cross or continue (Figure 2).

### 2.1 Railway Traffic Rules, topological and requirements constraints

A valid running map must satisfy and optimize the set of existing constraints in the problem. Some of the main constraints to be considered are:

1. **Traffic rules** guarantee crossing and overtaking operations. The main rules to take into account are:
   - ***Crossing constraint***: Any two trains ($T_i$ and $T_j$) going in opposite directions must not simultaneously use the same one-way track (A-B).

     $$T_i Arrives_A < T_j Departures_A \text{ or } T_j Arrives_B < T_i Departures_B$$

     The crossing of two trains can be performed only on two-way tracks and at stations, where one of the two trains has been detoured from the main track (Figure 2). Several crossings are shown in Figure 1.

- ***Overtaking constraint***: Any two trains ($T_i$ and $T_j$) going at different speeds in the same direction can only overtake each other at stations.

  $$T_i Departures_A < T_j Departures_A \rightarrow T_i Arrives_B < T_j Arrives_B$$

  The train being passed is detoured form the main track so that the faster train can pass the slower one (see Figure 2).

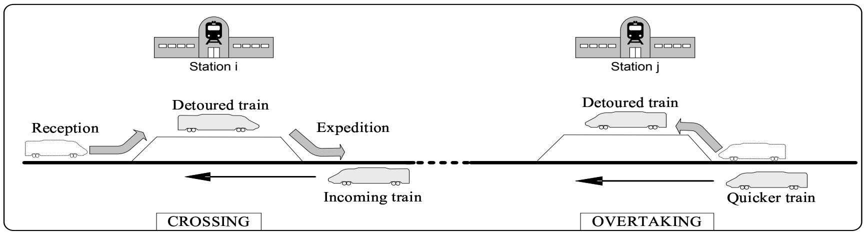


**Fig. 2.** Constraints related to crossing and overtaking in stations

- ***Expedition time constraint***. There exists a given time to put a detoured train back on the main track and exit from a station.
- ***Reception time constraint***. There exists a given time to detour a train from the main track so that crossing or overtaking can be performed.
- ***Succession time constraint***. Any two trains traveling in the same direction must respect the safety headway between trains (even if speeds are different). This should be maintained at arrival times, departures times and along the entire path. The succession time constrain depends on the features of traffic control in each section between two locations (manual, automatic, etc.)

2. **User Requirements**: The main constrains due to user requirements are:
   - ***Type and Number of trains*** going in each direction to be scheduled and ***Travel time*** between locations.
   - ***Path of trains***: Locations used and ***Stop time*** for commercial purposed in each direction.
   - ***Scheduling frequency***. The frequency requirements of the departure of trains in both directions This constraint is very restrictive, because, when crossing and overtaking are performed, trains must wait for a certain time interval at stations. This interval must be propagated to all trains going in the same direction in order to maintain the established scheduling frequency. The user can require a fixed frequency, a frequency within a minimum and maximum interval, or multiple frequencies.
   - ***Time interval*** for the departure of the first train going in one direction and the departure of the first train going in the opposite direction.
   - ***Maximum slack***. This is the maximum time allowed to perform all the technical operations.
3. **Topological railways infrastructure and type of trains** to be scheduled give rise other constraints to be taken into account. Some of them are:

- Number of ***tracks in stations*** (to perform technical and/or commercial operations) and the number of tracks between two locations (one-way or two-way). No crossing or overtaking is allowed on a one-way track,
- ***Closing times in the locations***, when no technical and /or commercial operations can be performed,
- Added ***time for the stopping and starting*** process of a train due to an unexpected/unscheduled technical stop.

In accordance with user requirements, the system should obtain the best solutions available so that all the above constraints are satisfied. Several criteria can exist to qualify the optimality of solutions: minimize duration and/or number of technical stops, minimize the total time of train trips (span) of the total schedule, giving priority to certain trains, etc.

### 2.2 General System Architecture

The general outline of our system is presented in figure 3. It shows several steps, some of which require the direct interaction with the human user to insert requirement parameters, parameterize the constraint solver for optimization, or modify a given schedule. First of all, the user should require the parameters of the railway network and the train type from the central database (Figure 4). This database stores the set of locations, lines, tracks, trains, etc. Normally, this information does not change, but authorized users may desire to change this information. With the data acquired from the database, the system carries out a *preprocessing step*, in which a linear constraint solver for optimization is used to solve a linearized problem in order to complete the formal mathematical model. In the *processing step*, the formal mathematical model is solved by a mixed-integer constraint solver for optimization, returning the running map data. If the mathematical model is not feasible, the user must modify the most restrictive problem parameters. If the running map is consistent, the graphic interface plots the scheduling (Figure 6). Afterwards, the user can graphically interact with the scheduling to modify the arrival or departure times. Each interaction is automatically checked by the constraint checker in order to guarantee the consistency of changes. The user can finally print out the scheduling, to obtain reports with the arrival and departure times of each train in each location, or graphically observe the complete scheduling topology.

## 3 Optimization Process: Preprocesses and Heuristics

The two main issues in our problem are (i) the specification of the model according to the existing constraints, and (ii) the constraint solver for optimization, which requires mathematical techniques, criteria and heuristics. This problem is more complex than job-shop scheduling [7], [10]. Here, two trains, traveling in opposite directions, as well as two trains traveling in the same direction use tracks between two locations for different durations, and these durations are

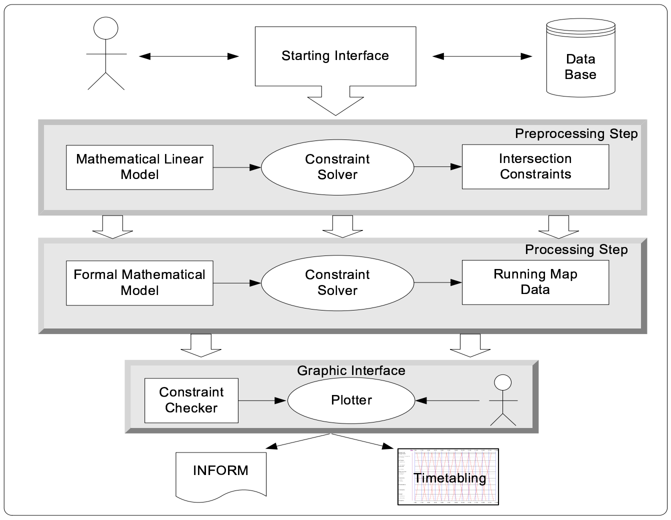


**Fig. 3.** General scheme of our tool.

causally dependent on how the scheduling itself is done (ie: order of tasks), due to the stopping, and starting time for trains in a non-required technical stop, expedition, reception, succession times, etc. Some processes (detour from the main railway) may or may not be required for each train at each location. In our system, the problem is modeled as a CSP, where finite domain variables represent frequency and arrival and departure times of trains of locations. Relations on these variables permit the management of all the constraints due to the user requirements, topological constraints, traffic rules, commercial stops, technical operation, maximum slacks, etc. Hundred of trains, of different types and speeds, in different directions, along paths of dozens of stations have to be coordinated. Thus, many variables, and many and very complex constraints arise. The problem turns into a mixed-integer programming problem, in which thousands of inequalities have to be satisfied and a high number of variables take only integer values. As is well known, this type of model is far more difficult to solve than linear programming models. Depending on the problem size, some pre-processes are performed in the system in order to reduce the number of variables and the complexity of the complete mathematical model:

1. Topological and geometrical processes. Identification of bottlenecks, periodicity of running maps, possible wide-paths for trains, etc.
2. Linear programming approach, in order to identify potential crossing or overtaking. Important decisions in the linear problem are the probabilistic feasibility of stations where crossing or overtaking can be performed, optimal initial departure times to minimize intersections, frequency, etc.

Afterwards, the optimization process is performed in several ways according to the level of required solutions and the problem size:

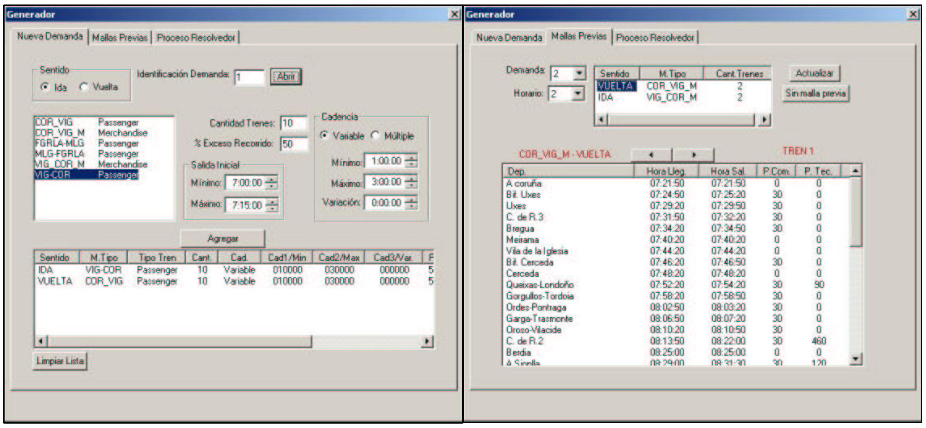
1. Complete: The process is performed taking into account the entire problem.
2. Incremental: The process performs an incremental coordination of trains.
3. Iterative: The solution is obtained by replicating a pattern found in a previous process of a simpler problem.

In addition, several heuristics and prioritization criteria are applied. These pre-processes as well as the level of heuristics applied in the optimization process can be selected by the user (Figure 5). However, the system can also automatically recommend or select the appropriate choices depending on different parameters and the complexity of the problem.

## 4 Functionalities and System Interfaces

In this section, we make a brief description about the functionalities and some interfaces of the developed tool:

1. Specify a demand (Fig. 4). Through several interfaces, the user provides data about requirements of a demand: Number and type of trains to be scheduled, path, direction, frequency, commercial stops, time interval for the initial departure of the first train, maximum slacks, previous demands to be considered on the same network, etc.



Sentido: ● Ida ○ Vuelta

Identificación Demanda: 1

| | |
|---|---|
| COR_VIG | Passenger |
| COR_VIG_M | Merchandise |
| FGRLA-MLG | Passenger |
| MLG-FGRLA | Passenger |
| VIG_COR_M | Merchandise |
| VIG-COR | Passenger |

Cantidad Trenes: 10

% Exceso Recorrido: 50

Salida Inicial
Mínimo: 7:00:00
Máximo: 7:15:00

Cadencia: ● Variable ○ Múltiple
Mínimo: 1:00:00
Máximo: 3:00:00
Variación: 0:00:00

| Sentido | M.Tipo | Tipo Tren | Cant. | Cad. | Cad1/Min | Cad2/Max | Cad3/Var. | F |
|---|---|---|---|---|---|---|---|---|
| IDA | VIG-COR | Passenger | 10 | Variable | 010000 | 030000 | 000000 | 5 |
| VUELTA | COR_VIG | Passenger | 10 | Variable | 010000 | 030000 | 000000 | 5 |

Demanda: 2
Horario: 2

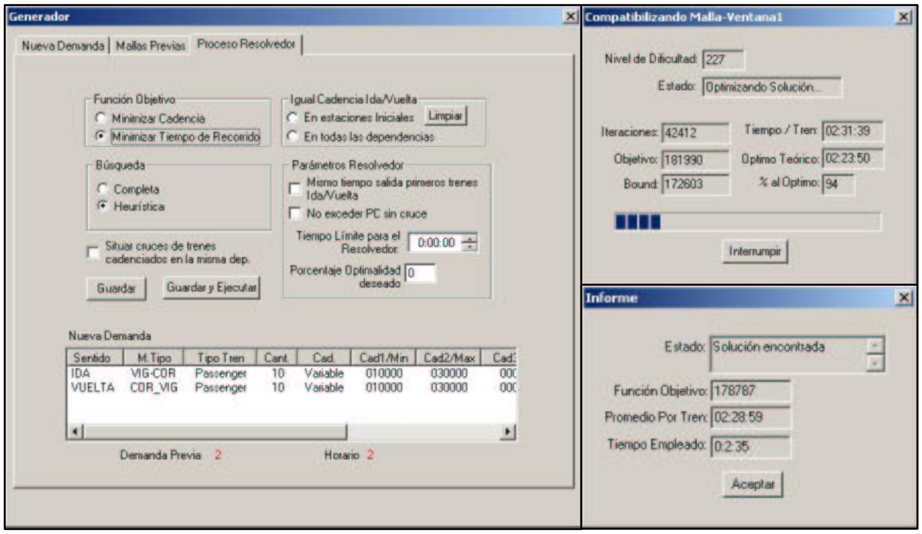
| Sentido | M.Tipo | Cant.Trenes |
|---|---|---|
| VUELTA | COR_VIG_M | 2 |
| IDA | VIG_COR_M | 2 |

COR_VIG_M - VUELTA

| Dep. | Hora Lleg. | Hora Sal. | P.Com. | P. Tec. |
|---|---|---|---|---|
| A coruña | 07:21:50 | 07:21:50 | 0 | 0 |
| Bif. Uxes | 07:24:50 | 07:25:20 | 30 | 0 |
| Uxes | 07:29:20 | 07:29:50 | 30 | 0 |
| C. de R.3 | 07:31:50 | 07:32:20 | 30 | 0 |
| Bregua | 07:34:20 | 07:34:50 | 30 | 0 |
| Meirama | 07:40:20 | 07:40:20 | 0 | 0 |
| Vila de la Iglesia | 07:44:20 | 07:44:20 | 0 | 0 |
| Bif. Cerceda | 07:46:20 | 07:46:50 | 30 | 0 |
| Cerceda | 07:48:20 | 07:48:20 | 0 | 0 |
| Queixas-Londoño | 07:52:20 | 07:54:20 | 30 | 90 |
| Gorgullos-Tordoia | 07:58:20 | 07:58:50 | 30 | 0 |
| Ordes-Pontraga | 08:02:50 | 08:03:20 | 30 | 0 |
| Garga-Trasmonte | 08:06:50 | 08:07:20 | 30 | 0 |
| Oroso-Vilacide | 08:10:20 | 08:10:50 | 30 | 0 |
| C. de R.2 | 08:13:50 | 08:22:00 | 30 | 460 |
| Berdia | 08:25:00 | 08:25:00 | 0 | 0 |
| A Sionlla | 08:29:00 | 08:31:30 | 30 | 120 |

**Fig. 4.** Some user requirement interfaces

2. Parameterization of optimization process. (see Figure 5 left). The optimization process can be parameterized to bound the search space and/or the execution time, to perform an incremental search, etc. During the solving process, using mathematical optimization packages, its state is displayed (Figure 5 upper right), and when it ends, the information about its final state is shown. (Figure 5 lower right).



Función Objetivo
○ Minimizar Cadencia
● Minimizar Tiempo de Recorrido

Igual Cadencia Ida/Vuelta
○ En estaciones Iniciales
○ En todas las dependencias

Búsqueda
○ Completa
● Heurística

Parámetros Resolvedor
☐ Mismo tiempo salida primeros trenes Ida/Vuelta
☐ No exceder PC sin cruce
Tiempo Límite para el Resolvedor: 0:00:00
Porcentaje Optimalidad deseado: 0

☐ Situar cruces de trenes cadenciados en la misma dep.

Nueva Demanda

| Sentido | M.Tipo | Tipo Tren | Cant. | Cad. | Cad1/Min | Cad2/Max | Cad |
|---|---|---|---|---|---|---|---|
| IDA | VIG-COR | Passenger | 10 | Variable | 010000 | 030000 | 000 |
| VUELTA | COR_VIG | Passenger | 10 | Variable | 010000 | 030000 | 000 |

Demanda Previa 2
Horario 2

Nivel de Dificultad: 227
Estado: Optimizando Solución...
Iteraciones: 42412
Objetivo: 181990
Bound: 172603
Tiempo / Tren: 02:31:39
Optimo Teórico: 02:23:50
% al Optimo: 94

Estado: Solución encontrada
Función Objetivo: 178787
Promedio Por Tren: 02:28:59
Tiempo Empleado: 0:2:35

**Fig. 5.** Some interfaces for the optimization process

## 5 Evaluation

The application and performance of this system depends on several factors: Railway topology (locations, distances, tracks, etc.), number and type of trains (speeds, starting and stopping times, etc.), frequency ranges, initial departure interval times, maximum slacks, etc. Several running maps are shown in this section. Figure 6 shows the schedule obtained from a requirement for an established frequency of only one type of train, in both directions. Thus, only crossing problems appear in one-way track sections and are performed in adjacent stations. Time slacks for technical operations due to crossing are minimized. The running path shown is optimal for the given parameters.

The system allows the user interactively modify an obtained running map (see Figure 6 lower right). A range between any two stations can be selected, so the user can modify the arrival and departure times between them, the commercial and/or technical stops, etc. Interactively, the system checks these changes to assure constraint fulfillment, so that non-valid running maps that are not in accordance with traffic rules, topological and train constraints are allowed. This feature permits the user to adapt running maps to special circumstances and to try alternative scheduling, etc.

Figure 7 (left) displays a running map of two different types of trains. Thus, two different requirements should be coordinated on the same rail network. Each requirement specifies a given number and type of train, frequency, start times, paths, location stops, etc. More specifically, a new requirement is added to the running map in Figure 6, so that crossing and overtaking problems appear. Therefore, the system allows the user to add new requirements to a previous

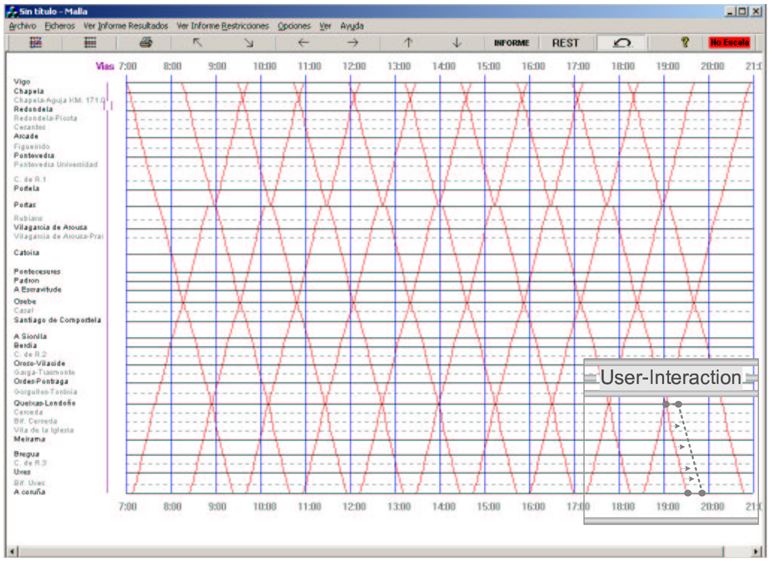
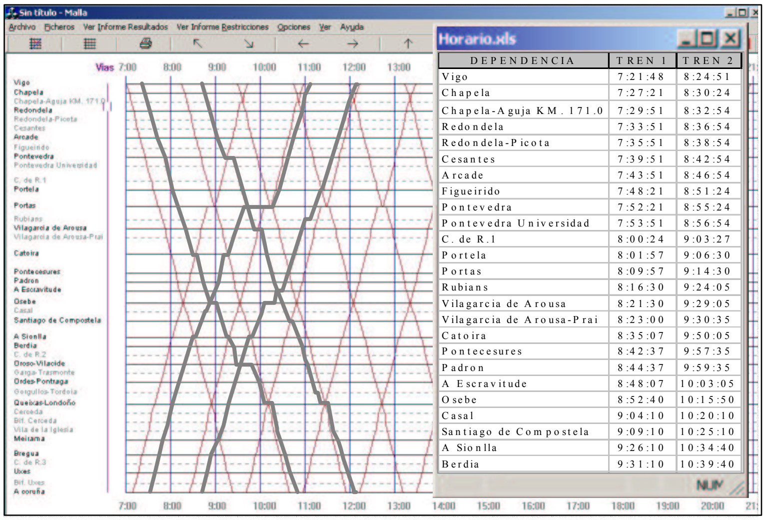


**Fig. 6.** A timetable with only one cadenced demand



| DEPENDENCIA | TREN 1 | TREN 2 |
|---|---|---|
| Vigo | 7:21:48 | 8:24:51 |
| Chapela | 7:27:21 | 8:30:24 |
| Chapela-Aguja KM. 171.0 | 7:29:51 | 8:32:54 |
| Redondela | 7:33:51 | 8:36:54 |
| Redondela-Picota | 7:35:51 | 8:38:54 |
| Cesantes | 7:39:51 | 8:42:54 |
| Arcade | 7:43:51 | 8:46:54 |
| Figueirido | 7:48:21 | 8:51:24 |
| Pontevedra | 7:52:21 | 8:55:24 |
| Pontevedra Universidad | 7:53:51 | 8:56:54 |
| C. de R.1 | 8:00:24 | 9:03:27 |
| Portela | 8:01:57 | 9:06:30 |
| Portas | 8:09:57 | 9:14:30 |
| Rubians | 8:16:30 | 9:24:05 |
| Vilagarcia de Arousa | 8:21:30 | 9:29:05 |
| Vilagarcia de Arousa-Prai | 8:23:00 | 9:30:35 |
| Catoira | 8:35:07 | 9:50:05 |
| Pontecesures | 8:42:37 | 9:57:35 |
| Padron | 8:44:37 | 9:59:35 |
| A Escravitude | 8:48:07 | 10:03:05 |
| Osebe | 8:52:40 | 10:15:50 |
| Casal | 9:04:10 | 10:20:10 |
| Santiago de Compostela | 9:09:10 | 10:25:10 |
| A Sionlla | 9:26:10 | 10:34:40 |
| Berdia | 9:31:10 | 10:39:40 |

**Fig. 7.** A timetable with two demands

running map so that this previous running map can be modified or can be kept the same, depending on the user's needs. The different lines in Figure 6 and in Figure 7 correspond to the paths performed by each train, respectively, where each point of a line corresponds to the position of a train at a given time.

## 6 Conclusions

We have presented the main features of a flexible and useful tool for solving and plotting optimized train schedules in collaboration with the National Network of Spanish Railways (RENFE). No other equivalent system is known from authors. This tool transforms railway problems into formal mathematical models. The NP-Hard complexity of problems of this kind requires that several heuristic criteria and pre-process be parameterized by the user or automatically selected by the application. They are then added to the solving process in order to bound the search space and to obtain an optimal solution.

The main features of the proposed approach are: (i) the access to databases to obtain centralized data about the rail network, trains and required paths, (ii) the model specification for a complex problem, (iii) the various parameterizations, levels and approaches for search processes for optimization, (iv) the graphical interfaces and interactivity with the user. This tool, at a current stage of pre-integration, supposes the application of methodologies of Artificial Intelligence in a problem of great interest and will assist railways managers in optimizing the use of railway infrastructures and will help them in the resolution of complex scheduling problems.

## References

1. Bussiecky, M.R., Winter, T., Zimmermann, U.T. *Discrete optimization in public rail transport*, Mathematical Programming 79(3), (1997), 415-444.
2. Bussiecky, M.R., Zimmermann, U.T. *Combinatorial Optimization Methods for Optimal Lines in Real-World Railway Systems*, Technical Report TR-95-03, (1996).
3. Caprara, A., Fischetti, M., Toth, P. *Modeling and Solving the Train Timetabling Problem*, Research Report OR/00/9 DEIS, (2000).
4. Caprara, A., Fischetti, M., Guida, P., Monaci, M., Sacco, G., Toth, P. *Solution of Real-World Train Timetabling Problems*, 34th Annual Hawaii International Conference on System Sciences ( HICSS-34) **3** (2001).
5. Chiu, C.K., Chou, C.M., Lee, J.H.M., Leung, H.F., and Leung, Y.W., *A Constraint-Based Interactive Train Rescheduling Tool*, Constraints **7** (2002), 167–198.
6. Kaas, A.H., *Methods to Calculate Capacity of Railways*, Ph. Dissertation (1998).
7. Kreuger, P.; Carlsson, M.; Olsson, J.; Sjoland, T.; Astrom, E. *The TUFF train scheduler: Two duration trip scheduling on multi directional track networks*. Workshop on Tools and Environments for Constraint Logic Programming, (1997).
8. Lindner, T., *Train schedule optimization in public rail transport*, Ph. Dissertation, Technische Universitat Braunschweig, Germany, (2000).
9. Oliveira, E., Smith, B.M., *A Job-Shop Scheduling Model for the Single-Track Railway Scheduling Problem*, Research Report 2000.21, University of Leeds, (2000).
10. Zuidwijk, R.A., Kroon, L.G., *Integer Constraints for Train Series Connections*, Erasmus Research Institute of Management (ERIM), Discussion Paper. (2002).