

## Research Article

# A New Approach to Shortest Route Finding in a Railway Network with Two Track Gauges and Gauge Changeovers

Alberto Almech <sup>1</sup>, Eugenio Roanes-Lozano <sup>2</sup>, Carmen Solano-Macías <sup>3</sup>,  
and Antonio Hernando <sup>4</sup>

<sup>1</sup>Universidad Complutense de Madrid, Madrid, Spain

<sup>2</sup>Instituto de Matemática Interdisciplinar & Depto. de Didáctica de las Ciencias Experimentales, Sociales y Matemáticas, Universidad Complutense de Madrid, Madrid, Spain

<sup>3</sup>Departamento de Información y Comunicación, Universidad de Extremadura, Badajoz, Spain

<sup>4</sup>Depto. de Sistemas Inteligentes Aplicados, E.T.S.I. de Sistemas Informáticos, Universidad Politécnica de Madrid, Madrid, Spain

Correspondence should be addressed to Eugenio Roanes-Lozano; [eroanes@mat.ucm.es](mailto:eroanes@mat.ucm.es)

Received 16 June 2019; Revised 17 September 2019; Accepted 23 September 2019; Published 5 December 2019

Academic Editor: Libor Pekař

Copyright © 2019 Alberto Almech et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The Spanish railway network is a complex one, due to the existence of *standard gauge* (1,435 mm), *Iberian gauge* (1,668 mm), and *dual gauge* (with three rails) subnetworks, as well as connections between the two first subnetworks (at the so-called *gauge changeovers*), usable by *dual gauge trains*. Two of the authors developed a package for shortest route finding, and consumes, costs, and emissions calculations in the Spanish railway network (within the frame of two research projects funded by the *Fundación de los Ferrocarriles Españoles–Spanish Railways Foundation*). Nevertheless, the approach required from a curious specific algorithm to be developed ad hoc. In this paper, a much simpler approach to shortest route finding in such a network, which merges the three subnetworks into two and considers the gauge changeovers as connections between the two new subnetworks, is detailed. This approach allows to use any standard shortest route algorithm. It has been implemented in the computer algebra system *Maple* and is applied in this paper both to small specific cases and to the whole Spanish railway network.

## 1. Introduction

*1.1. About the Previously Developed Package RutasOptiRed and Its Algorithm.* Shortest route finding in a railway network is a topic where two of the authors have worked for several years. In the frame of two research projects funded by the *Fundación de los Ferrocarriles Españoles (Spanish Railways Foundation)*, they developed from scratch in 2010 and 2011 a complex piece of software (denoted *RutasOptiRed*). The first version can find shortest routes and evaluate with high precision timings for the different pieces of rolling stock of *Renfe Operadora* (Spanish Railways operator) running on *Adif* (*ADministrador de Infraestructuras Ferroviarias*, that is, *Spanish Railway Infrastructure Administrator*) railway network [1]. The second version adds to this the possibility to detail the electric and Diesel consumes, CO<sub>2</sub> emissions, and costs [2]. An example of application can be found in [3].

In order to achieve a high precision, *RutasOptiRed* requires details of all parts of the sections in the railway network, as well as of all the different pieces of rolling stock, to be given as input (in order to obtain precise results). Each edge of the undirected weighted graph (note that as all the graphs considered hereinafter are *undirected weighted graphs*, so we shall refer to them as *graphs* for the sake of brevity) has a list of characteristics associated as follows: track gauge, maximum speed, etc.

It even has an appropriate GUI that allows to comfortably introduce the data of each scenario (Figure 1).

*Adif* network includes two track gauges, some sections with three rails (dual gauge) and several gauge changeovers (what will be detailed in Section 1.3). Such a network required *RutasOptiRed* to use a curious specific algorithm to be developed ad hoc. It is a complex and unintuitive adaptation of Dijkstra's algorithm:

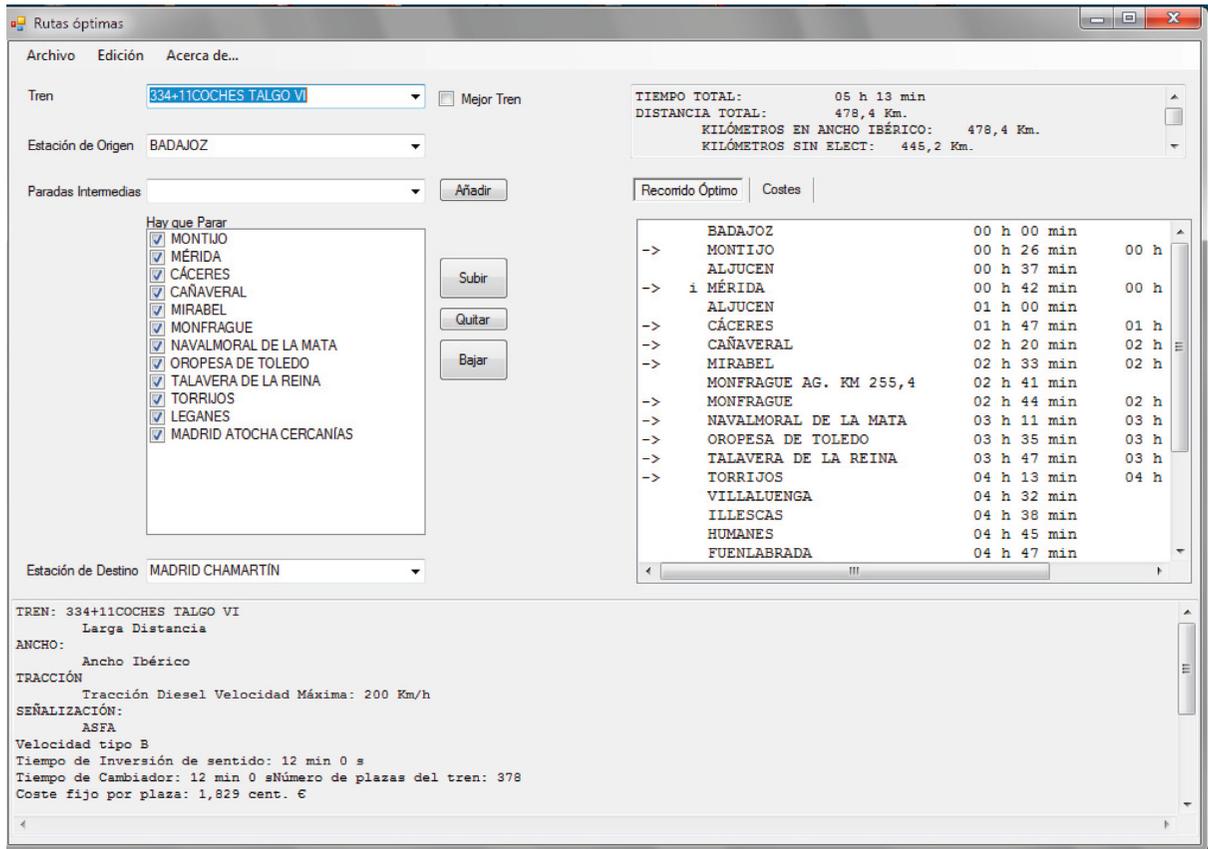


FIGURE 1: Example of data introduction to *RutasOptiRed*: the user is asking to perform the computations corresponding to a Class 334 locomotive hauling 11 *Talgo VI* alloy coaches in the Badajoz–Madrid Chamartín route.

- (i) A different graph is built for each class of trains (!)
- (ii) The track gauge the train is using at each moment is considered
- (iii) The vertices represent the sections between stations, and the edges represent the possibility of that class of trains to pass from one section to the other (!), according to the track gauge it is using at that moment.

This latter algorithm is convoluted and is detailed in a separate paper [4].

*1.2. About the New Approach Introduced in This Article.* In the present paper, a much simpler approach to shortest route finding in such a complex network is introduced:

- (i) It merges the three subnetworks (*standard gauge*, *Iberian gauge*, and *dual gauge*) into two *imaginary* (in the sense that they do not correspond to any existing subnetwork) ones
- (ii) It considers the gauge changeovers as connections between the two new subnetworks

The main advantage of this new approach is that vertices and edges are assigned the usual way. Consequently, any standard shortest route algorithm can be used.

Based on the new approach, a *Maple* (*Maple* is a trademark of *Waterloo Maple Inc.*) package that allows to explore different alternatives in any railway network with two different track gauges and gauge changeovers has been developed.

The work presented here is an improved and extended version of a work presented at CMMSE'2017 conference [5].

*1.3. The Complex Spanish Railway Network: A Brief Overview.* As said above, *Adif* is in charge of the Spanish railway network, meanwhile *Renfe Operadora* runs the passengers trains.

The Spanish railway network is particularly complex [6], because it has tracks of two gauges: *standard gauge* (1,435 mm) and *Iberian gauge* (broad gauge: 1,668 mm) as well as *dual gauge* sections (Figure 2).

*Iberian gauge* was the choice for the Spanish main network in the XIXth Century, but all the new high-speed lines since the first one (the 1992 Madrid–Seville) have been built in *standard gauge*. Note that some isolated sections of the high-speed network have been provisionally opened in *Iberian gauge* using dual gauge concrete sleepers (Ourense–Santiago section of the NW high-speed line) or are planned to be provisionally opened this way (sections within Extremadura region of the Madrid–Badajoz–Lisbon high-speed line) (see Figure 2).

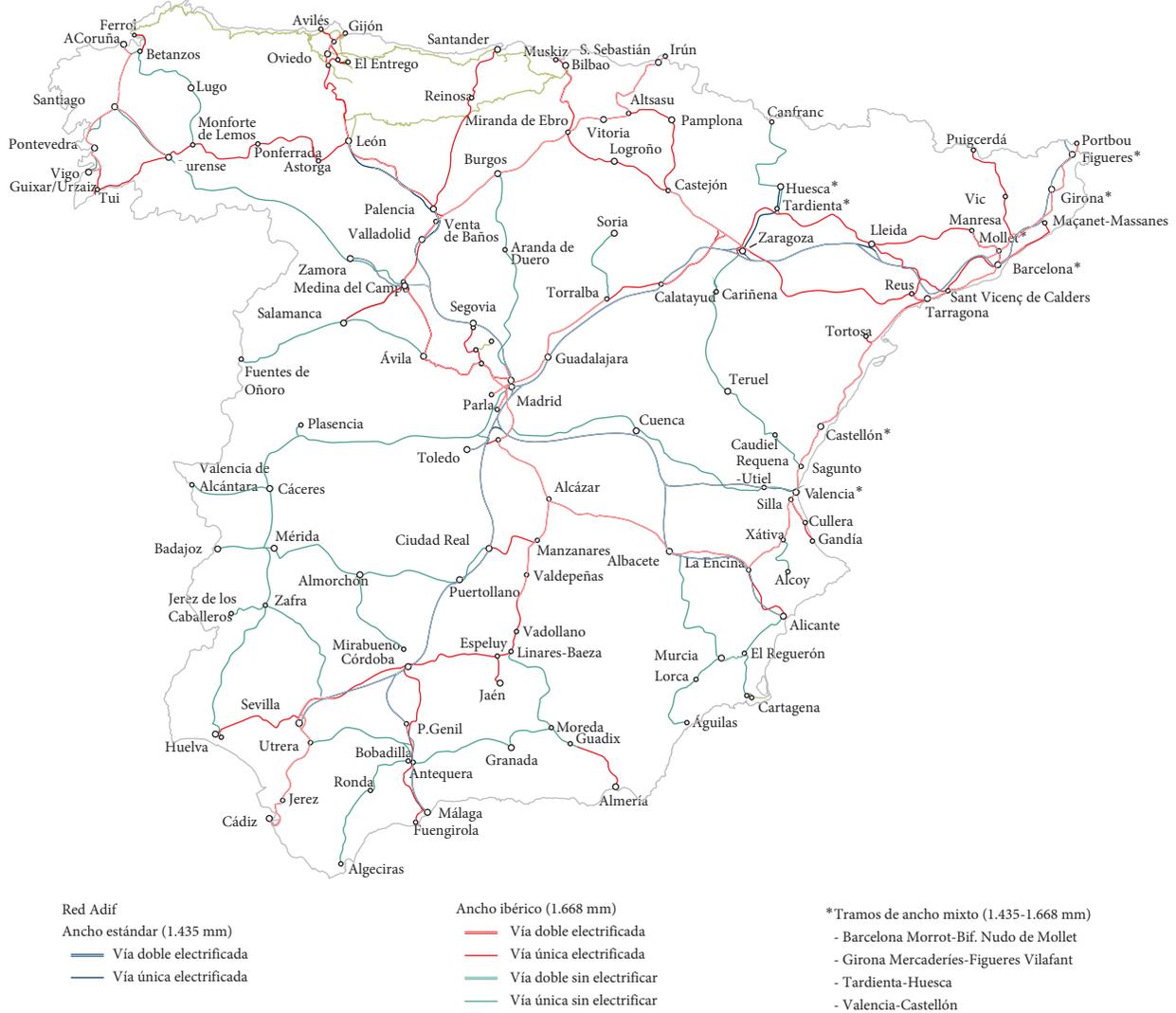


FIGURE 2: The Spanish Iberian and standard gauge railway network (2018) (courtesy of the *Fundación de los Ferrocarriles Españoles (Spanish Railways Foundation)*). The legend of the map is translated afterwards. First column: standard track gauge (1,435 mm); electrified double track; electrified single track. Second column: Iberian track gauge (1,668 mm); electrified double track; electrified single track; nonelectrified double track; nonelectrified single track. Third column: Dual track gauge sections (1,435–1,668 mm) (they are enumerated afterwards).

*Dual gauge* (also denoted *mixed gauge*) sections have three rails [7], one common rail on one side of the sleepers and two rails on the other side. Trains of the narrower gauge use the common rail and the closest rail. Trains of the wider gauge use the common rail and the furthest rail.

In the beginning of 2019, the dual gauge sections are the Tardienta–Huesca section of the Canfranc line (South of the Central Pyrenees) and sections along the so-called *Mediterranean corridor*: a long section North of Valencia, a bypass of Barcelona and some sections North of Barcelona, between this city and the French border (see *Mapa 6* in [8]). More are planned or under work.

Moreover, the *Iberian gauge* and *standard gauge* subnetworks are not isolated. At some locations, there are gauge

changeovers [9, 10] that connect both subnetworks (their present location can also be found in *Mapa 6* in [8]).

Note that *Renfe* owns some classes of dual gauge locomotive hauled trains and multiple units [11]. Since 1969 till the beginning of the XXIst Century, locomotive hauled *Talgo* trains crossed the French border toward Paris and Geneva (Portugal has the same track gauge as Spain). This was no longer necessary when the Spanish standard gauge network reached the French border, and gauge changeovers are now situated within Spain.

Nowadays, there are not only dual gauge coaches but only whole train sets, like the *Talgo* pushed-pulled classes 130 and 730 (the latter is hybrid and can work in both electric and Diesel modes) (Figure 3) and multiple units like CAF classes 120 and 121 (Figure 4).



FIGURE 3: A Talgo Class 130 push-pull dual gauge bivoltage train of Renfe rolling stock. Class 730 is identical except that it incorporates an electrical power generator wagon immediately after each locomotive.



FIGURE 4: A CAF Class 121 dual gauge bivoltage multiple unit of Renfe rolling stock.

The narrow gauge network (1,000 mm) is nowadays run by Adif, after absorbing FEVE (*Narrow Gauge Railway Company*) on December 31, 2012. Nevertheless, the 1,000 mm network is not considered here, as it is isolated from the 1,435 mm and 1,668 mm networks, and there is no rolling stock that can run on narrow gauge tracks and also on tracks of any of the other two gauges.

**1.4. Related Works.** There are many issues regarding railway operation: planning, timetabling, scheduling, etc., (see, e.g., [12–14]) or line planning [15], just to mention a few, but they consider *normal* railway networks (that is, railway networks of a certain unique track gauge).

The goal of our work is shortest route finding in a railway network with two track gauges. There are many works on shortest route finding applied to road traffic [16, 17], railway networks [18–21], transportation in general [22], and accessibility considering times [19], but we know of no similar work dealing with railway networks with two track gauges and gauge changeovers (except the already mentioned *RutasOptiRed*).

**1.5. The Curious Madrid–Canfranc Counterexample.** Let us analyze an example of what makes impossible to directly use a shortest route algorithm in a naive approach to the Spanish

railway network (this problem arose when testing the first version of *RutasOptiRed*).

Let us consider the Madrid–Zaragoza–Tardienta–Huesca–Canfranc route of Adif railway network.

There are two railway connections between Madrid and Zaragoza: one is the classic Iberian gauge line and the other is part of the Madrid–Barcelona high-speed line (in NE Spain).

Meanwhile, Zaragoza–Tardienta–Huesca–Canfranc is a branch line in Central Pyrenees (Figure 2). Its Tardienta–Huesca section was chosen to test new technical solutions such as dual gauge (three rails) tracks and dual voltage catenary (commutable between 3,000 V DC and 25,000 V AC, according to the kind of train traversing the section). Note that classic lines are electrified at 3,000 V DC and high-speed lines at 25,000 V AC.

Summarizing, the situation of this peculiar route is the following:

- (i) There are gauge changeovers at Madrid Chamartín and Madrid Atocha railway stations (the two main Madrid stations)
- (ii) There are international gauge and Iberian gauge lines: Madrid–Zaragoza–Tardienta
- (iii) There is a gauge changeover at Zaragoza Delicias station, but no one further toward Canfranc
- (iv) The Tardienta–Huesca is a dual gauge section
- (v) The Huesca–Canfranc section is an Iberian gauge one

This route and its characteristics are represented in a simplified way in Figure 5.

In this case, if we ask for the shortest route between Madrid and Canfranc, a standard shortest route algorithm will try to reach the intermediate stations (Zaragoza and Tardienta) as fast as possible. For a variable gauge train, we would obtain the following routes:

- (i) Madrid–Zaragoza: train set to standard gauge, along the high-speed line
- (ii) Zaragoza–Tardienta: train set to standard gauge, along the high-speed line
- (iii) Tardienta–Huesca: train set to standard gauge, along the dual gauge section

But then, the train would reach Huesca set as a standard gauge train, and there is no gauge changeover at Huesca, so the train could not go on (the rest of the route is an Iberian gauge section)!

Then, the surprising solution given by a standard shortest route algorithm would be to return to the (far away) gauge changeover at Zaragoza and to return back to Huesca on Iberian gauge!

Summarizing, the existence of a dual gauge section (with three rails)

- (i) that is not at any of the ends of the route
- (ii) that has not got gauge changeovers at their endpoints

makes impossible to use a classic shortest route algorithm such as Dijkstra's [23] and Floyd's [24] (that is, algorithms

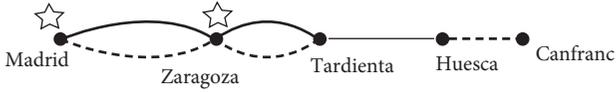


FIGURE 5: Between Madrid and Zaragoza and between Zaragoza and Tardienta, there is a high-speed line of standard gauge (thick line) and an Iberian gauge line (dashed line). There are gauge changeovers at Madrid and Zaragoza (stars). Between Tardienta and Huesca, there is a dual gauge section (thin line). Between Tardienta and Huesca, there is only an Iberian gauge line (dashed line).

that consider that the piece of rolling stock can move from any section of the network or subnetwork to another, without taking into account how the gauge of a variable gauge train is set).

## 2. The New Approach: Key Idea

As the problem is shortest route finding in a dual gauge railway network, we consider that the trains can run along tracks of both gauges and can pass through gauge changeovers (in order to take advantage of possibilities of the whole railway network). If only Iberian gauge or standard gauge trains were to be considered, it would be enough to prune the graph accordingly.

The key idea in the new approach presented here is to make up a fictitious new graph (from the three initial subnetworks of standard/Iberian/dual gauge) that incorporates all the track gauge information, but without associating track gauge details to its edges.

From a more visual point of view, the part of the railway infrastructure that can be traversed by trains of each of the two track gauges occupies a level or layer in the form of a subgraph (note that these two graphs do not correspond to any existing subnetwork). These levels are connected to each other (at certain vertices) by means of (artificially added) edges that correspond to the gauge changeovers.

Introducing these two layers will require to consider two vertices for each station that can be reached by trains of the two different track gauges (as will be detailed in Section 2.1).

This way any standard shortest route algorithm (such as Dijkstra's [23] and Floyd's [24]) can be used.

Let us detail the new approach afterwards.

**2.1. The New Approach: Notation and Initial (Given) Graphs.** Let  $H$  be a graph. We shall denote by  $V(H)$  the set of vertices of  $H$  and by  $E(H)$  the set of edges of  $H$ .

Let  $S$  be the set of stations of the railway network. The names of these stations should not have subscripts.

As expected, three graphs are initially considered (and must be given as input). As usual, but unlike *RutasOptiRed* [1, 2, 4], the vertices of the graphs represent the stations and the edges of the graphs correspond to the sections of the railway network. These graphs are as follows:

- (i)  $G_i = (V_i, E_i)$ : graph corresponding to the Iberian gauge railway network

- (ii)  $G_s = (V_s, E_s)$ : graph corresponding to the standard gauge railway network
- (iii)  $G_d = (V_d, E_d)$ : graph corresponding to the dual gauge railway network

Here,  $V_z$  ( $z \in \{i, s, d\}$ ) are the sets of vertices (stations) and  $E_z$  ( $z \in \{i, s, d\}$ ) are the sets of edges (sections of railway line) of the Iberian/standard/dual gauge railway networks (respectively). The weight associated to each edge is the corresponding timing (introduced directly or through length and average speed).

**2.2. The New Approach: Making up a Fictitious New Graph,  $G^*$ , Containing all the Information.** Let us begin with the initial graphs  $G_i$ ,  $G_s$ , and  $G_d$  described in Section 2.1.

The process to be carried out in order to build the new graph,  $G^*$ , can be summarized as follows:

- (i) If a station can be reached by trains of the two track gauges (either along normal tracks or dual gauge tracks), it will be described by two different vertices.

For instance, Zaragoza main station (*Estación de Zaragoza Delicias*) has tracks of both track gauges, and we shall consider two vertices:  $Zaragoza_i$  and  $Zaragoza_s$  (standing for *Zaragoza Iberian gauge* and *Zaragoza standard gauge*, respectively), even if there is only one real station.

More formally, let us denote

$$B = (V(G_i) \cap V(G_s)) \cup V(G_d). \quad (1)$$

For all vertices  $v \in B$ , two vertices,  $v_i$  and  $v_s$ , will be considered instead of the original  $v$ .

- (ii) Let us denote

$$\begin{aligned} B_i &= \{v_i \mid v \in B\}; \\ B_s &= \{v_s \mid v \in B\}. \end{aligned} \quad (2)$$

- (iii) Let us name by  $C_i \subseteq B_i$  and  $C_s \subseteq B_s$  the sets of stations where a gauge changeover is installed.

Obviously  $C_i$  and  $C_s$  consist of the same names with different subscripts, that is,

$$station_i \in C_i \iff station_s \in C_s. \quad (3)$$

as gauge changeovers connect the standard gauge and Iberian gauge tracks of the same station.

- (iv) Let us represent by

$$G_z|_{S_y}, \quad z \in \{i, s, d\}, y \in \{i, s\}. \quad (4)$$

the graph obtained by exchanging in the initial graph  $G_z = \{V_z, E_z\}$  the name of all vertices,  $v$ , such that  $v_y \in B_y$ , by  $v_z$ .

This way four new graphs are obtained:

- (a)  $G_i|_{S_i}$ , corresponding to the Iberian gauge subnetwork, with some vertices with an added subscript “i,”
  - (b)  $G_d|_{S_i}$ , corresponding to the dual gauge subnetwork, with some vertices with an added subscript “i,”
  - (c)  $G_s|_{S_s}$ , corresponding to the standard gauge subnetwork, with some vertices with an added subscript “s,”
  - (d)  $G_d|_{S_s}$ , corresponding to the dual gauge subnetwork, with some vertices with an added subscript “s.”
- (v) Now two new graphs can be constructed, corresponding to the accessibility in each track gauge, respectively:

$$\begin{aligned} G_i^* &= G_i|_{S_i} \cup G_d|_{S_i}, \\ G_s^* &= G_s|_{S_s} \cup G_d|_{S_s}. \end{aligned} \quad (5)$$

Note that  $G_i^*$  and  $G_s^*$  correspond to the sections of the network that can be traversed by an Iberian gauge train/standard gauge train, respectively. Let us recall that the names of stations with tracks of both track gauges, like *Zaragoza*, have been substituted by the names with *i* and *s* subscripts, like *Zaragoza<sub>i</sub>* and *Zaragoza<sub>s</sub>*, accordingly.

- (vi) Finally, the two previous graphs are merged in a single graph,  $G^*$  (Figure 6), and the connectivity between them is introduced by giving the gauge changeover positions (the edges corresponding to the gauge changeovers are of the form  $\{Zaragoza_i, Zaragoza_s\}$ ). More formally,

$$G^* = (V^*, E^*), \quad (6)$$

where

$$\begin{aligned} V^* &= V(G_i^*) \cup V(G_s^*), \\ E^* &= E(G_i^*) \cup E(G_s^*) \cup \{\{v_i, v_s\} \mid v_i \in C_i \wedge v_s \in C_s\}. \end{aligned} \quad (7)$$

**2.3. The New Approach: Remark.** The goal is to compute the shortest possible route for a given piece of rolling stock. Therefore, it will be supposed that an appropriate class of trains will be chosen by the end user (either standard gauge train, Iberian gauge train or dual gauge train). For instance, it is impossible to reach Cáceres with a class 103 train because this is a standard gauge train and this station is only in the Iberian gauge network.

Moreover, the piece of rolling stock chosen should be able to take advantage of the maximum speeds of the different sections of the route in order to obtain correct results.

Observe that the goal of the new piece of software described in this paper is only one of the goals of *RutasOptiRed*, but the

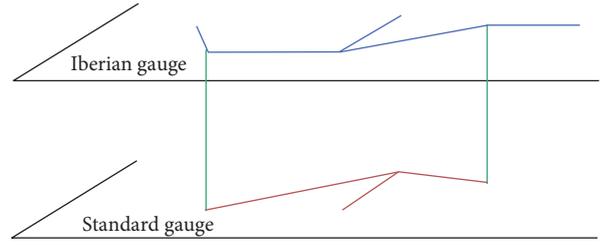


FIGURE 6: A 3D representation of graph  $G^*$  capturing the global accessibility of a certain railway network with two gauges. The two planes (layers) represented are  $z = 0$  (lower) and  $z = 1$  (upper). The  $z$ -axis is, as usual, the vertical axis. Graph  $G_i^*$  is represented in the upper plane, and its edges are drawn in blue. Graph  $G_s^*$  is represented in the lower plane, and its edges are drawn in red. Therefore, the stations with Iberian gauge tracks are in the upper plane ( $z = 1$ ) and their names have an “i” subscript or no subscript at all. Meanwhile, all the stations with standard gauge tracks are in the lower plane ( $z = 0$ ) and their names have an “s” subscript at all. The  $x$  and  $y$  coordinates of the stations of the forms  $v_i$  and  $v_s$  (like *Zaragoza<sub>i</sub>* and *Zaragoza<sub>s</sub>*) are identical. The vertical connections (in green) between the two horizontal planes represent the location of the gauge changeovers (that only sometimes exist in stations with tracks of both track gauges).

main one. The new approach that merges the dual gauge subnetwork with the Iberian and standard gauge networks allows to use a much simpler coding and algorithm.

**2.4. The New Approach: Advantages.** In the new approach, only either the real or theoretical average speed and length or the scheduled timing have to be assigned to each edge of  $G^*$ . As the edges correspond to sections of railway line or to gauge changeovers, origin to destination timings can be easily computed (just adding the timings of the sections along the route).

In the example described in Section 1.5 (Madrid–Canfranc line), the new approach proposed in this paper would consider the graph of Figure 7. It is clear that the error mentioned in Section 1.5 cannot arise, and the route chosen would be the following:

- (i) Madrid–Zaragoza (train set to standard gauge, along the high-speed line, because the speed is higher than in the Iberian gauge line)
- (ii) Gauge change at Zaragoza
- (iii) Zaragoza–Tardienta (train set to Iberian gauge, along the Iberian gauge line)
- (iv) Tardienta–Huesca (train set to Iberian gauge, along the dual gauge section)
- (v) Huesca–Canfranc (train set to Iberian gauge, along the Iberian gauge line)

Because when at Zaragozas, the only way to go on to Canfranc is to move to the upper layer, which can be reached at Zaragoza through the gauge changeover.

**2.5. The New Approach: A Final Remark.** Let us observe that the origin (respectively, destination) of the route can be stations with tracks of the two track gauges (such as Madrid in

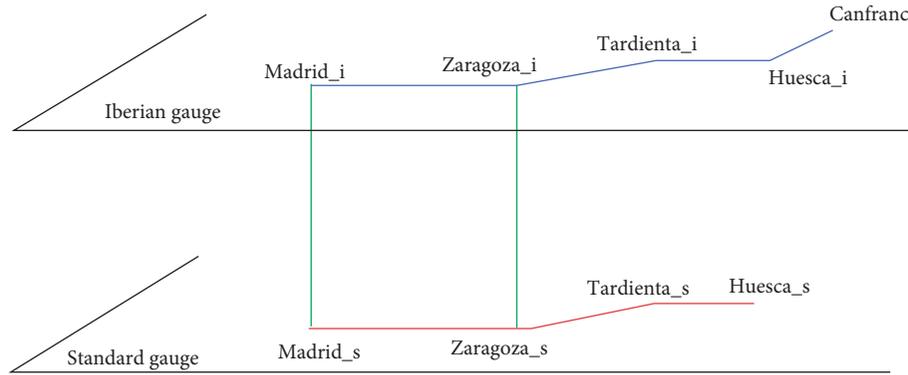


FIGURE 7: 3D representation in the new approach of the graph  $G^*$  corresponding to the Madrid–Canfranc routes. The five stations in the upper layer have Iberian gauge tracks. All except Canfranc have standard gauge tracks too (so they appear in the lower layer too). There are vertical connections at Madrid and Zaragoza because there are gauge changeovers installed at these stations.

Figure 7). In such case, the end user should choose the track gauge the train is set to when departing (respectively, arriving).

In case a wrong track gauge choice was made by the end user at an origin or destination station with gauge changeovers, the package would ask for passing through the gauge changeover there first, what is trivial to correct.

### 3. Implementation

The new package is implemented in *Maple*, using its *GraphTheory* package. This way the implementation is brief, only a few pages long. Details are avoided for the sake of space. The code is available from the authors.

Dijkstra’s algorithm is used because it is the one offered by *Maple* for this purpose. Nevertheless, this is just an example. The new approach presented in this article can be implemented using any shortest route algorithm in any appropriate software.

3.1. *Data Introduction 2D and 3D Graph Plotting.* Let us detail the steps of the data introduction to the present implementation and its possibilities.

Step 1: firstly, *Maple’s GraphTheory* and *plots* packages must be loaded:

```
with(GraphTheory):
```

```
with(plots):
```

Step 2: in order to produce an approximate plot, the package begins by defining a bidimensional array of possible locations for the stations, that is, the vertices of  $G_i$ ,  $G_s$ , and  $G_d$ . So, the second step consists in declaring the number of rows and columns of nodes in the rectangular lattice where the stations have to be allocated (they will have to be manually allocated, when introducing the edges of the railway networks, in positions that approximately correspond to their geographical coordinates). For example,

```
rows:=8:
```

```
columns:=12:
```

Step 3: afterwards, the code of the package presented here can be loaded:

```
read('RouteDual04.mpl');
```

Step 4: the sections of each of the three railway networks (edges of the graphs) have to be given through their end points, lengths and average speeds of the different sections. The package obtains the times per section (for each track gauge) from the distances and speeds. For instance,

```
introdato(nodo(3, 3), "Los Rosales", nodo(4, 3), "Lora", 20, 160):
```

introduces an Iberian gauge section.

Meanwhile,

```
t:=nodes:
```

```
introdato(t+nodo(6, 5), "C. Real_s", t+nodo(5, 5), "Puertollano_s", 39, 270):
```

introduces a standard gauge section

The dual gauge sections require a slightly more complex data introduction as their vertices have to define two virtual stations (one for each gauge). An example of introduction of the data of a section is

```
d:=2*nodes:
```

```
introdato(d+nodo(12, 13), "Huesca", d+nodo(12, 12), "Tardienta", 22, 160):
```

```
stations[nodo(12, 13)] := "Huesca_i":
```

```
stations[nodes+nodo(12, 13)] := "Huesca_s":
```

```
stations[nodo(12, 12)] := "Tardienta_i":
```

```
stations[nodes+nodo(12, 12)] := "Tardienta_s":
```

These sets of commands should be followed by a final `generate_A(i)`;

```
generate_timings_A(i);
```

where  $i = 0, 1, 2$  correspond to the Iberian, standard and dual gauge networks.

Step 5: the graphs of the different track gauges showing times and speeds can be plotted immediately after generating the corresponding  $A(i)$  by typing:

```
display(A);
```

(In Figure 8, note that the speeds have been obtained from the *Mapa 8* of the *Anexo H* of the official document [25], but these are very optimistic if taken as average speeds, because they do not consider partial

speed limitations along the section. Anyway, we shall consider them as illustration.)

Step 6: it is possible to erase the unused vertices in the timings graphs (thanks to an appropriate procedure developed ad hoc). The pruned graph can be obtained by typing

```
display(AT);
```

instead (in Figure 9 the timings of the Iberian gauge subnetwork in SW Spain) are shown).

Step 7: then, the timing to pass a gauge changeover and their locations in the lattice have to be introduced. For example,

```
gauge_changeovers(10,{15, 17, 41, 70, 84, 90});
```

Step 8: the real names of the nodes (stations) of the two layers can be shown instead of the numbers of the nodes (if desired), in order the end user to comfortably manipulate the package. It is enough to type:

```
names(0);
```

Step 9: a 2D representation of the whole network, where the three initial graphs have been merged into two,  $G_i^*$ ,  $G_s^*$ , as explained above, that also shows the gauge changeovers can be obtained by typing (see Figure 10):

```
display(IT);
```

Step 10: finally, a 3D representation of the whole network (in two levels) can be obtained by typing (see Figure 11):

```
nice_3D();
```

```
TwoLevels3DPlot;
```

Observe that how to represent a certain route can be found in Section 4.

**3.2. Obtaining the Best Route and Timing.** The analysis of the Madrid--Seville route is included as illustration (here, we suppose that the input data of the network have already been introduced). If, for example, the end user types:

```
Route_M_S:=bestRoute(H,"Madrid At_s","Sevilla_s");
```

the following output is obtained:

```
[[ "Madrid At_s", "C. Real_s", "Puertollano_s", "Cordoba_s", "Sevilla_s" ], 122]
```

that is, the best route and the corresponding timing (in minutes) have been stored in variable `Route_M_S`. The best route is a list of stations. They can be shown by typing:

```
Route_M_S[1];
```

```
["Madrid At_s", "C. Real_s", "Puertollano_s", "Cordoba_s", "Sevilla_s"]
```

```
Route_M_S[2];
```

```
122
```

Let us underline that the best timing (a Madrid--Seville nonstop) is nowadays  $2\text{ h }20\text{ m} = 140\text{ m}$ . The approximation is not bad, as

- (i) It has to be taken into account that high-speed trains in Spain are scheduled with a margin of a few minutes (due to a strict money back policy if there are small delays)
- (ii) The first kilometers in the neighborhood of the origin and destination stations are traversed at limited speed
- (iii) Not all intermediate stations are passed at full speed (details that are not considered by the package).

#### 4. Case Study: A Nontrivial Example in the Whole Spanish Railway Network

The package has been used to analyze the Madrid--Seville, Madrid--Badajoz and Badajoz--Lisbon routes, based in present and historically available lines [26--28].

It is clear that the accuracy of the timings obtained strongly depends on the accuracy of the input data.

In order to show the possibilities of the new package when treating bigger networks, we shall consider afterwards the whole Spanish railway network, excluding the narrow gauge subnetwork (see Figure 12). Note that all lines with passengers services are considered, but not all stations (only the main ones and all the junctions, in order to declare all the sections of the network considered).

Let us compute a nontrivial example in the whole network: the best route from Valencia (in the West coast) and Canfranc, (in the NW). Once the data have been introduced, and the graphs have been computed (see Section 3.1), it is very simple to ask the system or a best route and timing. To store it, for example, in variable `Route_V_C`, it is enough to type:

```
Route_V_C:=bestRoute(H,"Valencia_s","Canfranc");
Route_V_C=[[ "Valencia_s", "Requena_s", "Bif. Albacete_s", "Cuenca_s", "Madrid At_s", "Guadalajara_s", "Calatayud_s", "Zaragoza_s", "Tardienta_s", "Tardienta_i", "Huesca_i", "Canfranc" ], 237]
```

`Route_V_C` is a list with two elements: its first element is a list detailing the best route found and its second element is the estimated time (237 minutes in this case).

*Maple* took 0.015 seconds to perform this computation in a standard computer.

Afterwards a 2D diagram of the route, distinguishing when the train is set to Iberian or to standard gauge, can be asked by typing

```
PlottingRoute2d(H, Route_V_C);
```

```
PlotRoute2d;
```

and the plot of Figure 13 is obtained (let us recall that there are dual gauge sections, so how the train is set is the most precise information).

Finally, a 3d diagram of the route, that allocates the sections when the train is set to standard gauge in the upper plane and to Iberian gauge on the lower plane, can also be constructed just by typing

```
PlottingRoute3d(H, Route_V_C, {});
```

```
PlotRoute3d;
```

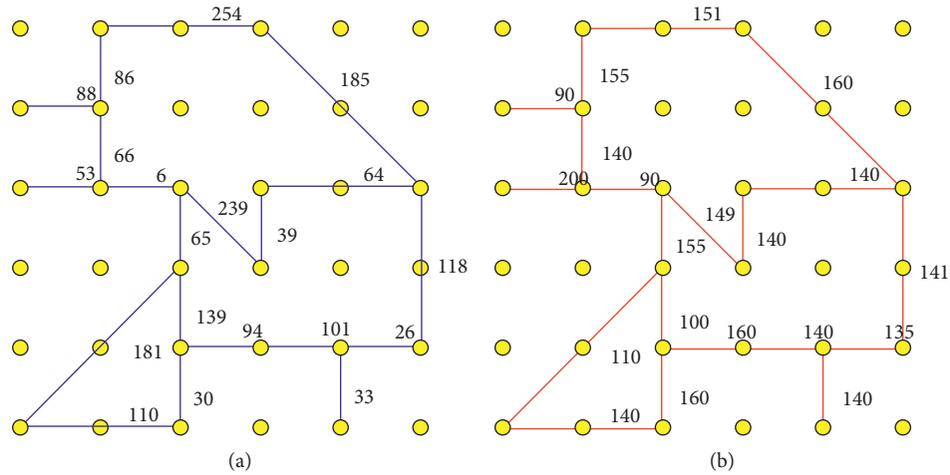


FIGURE 8: Diagrams of the Iberian gauge railway network of SW Spain. In this figure, distances are shown on the left and theoretical speeds on the right.

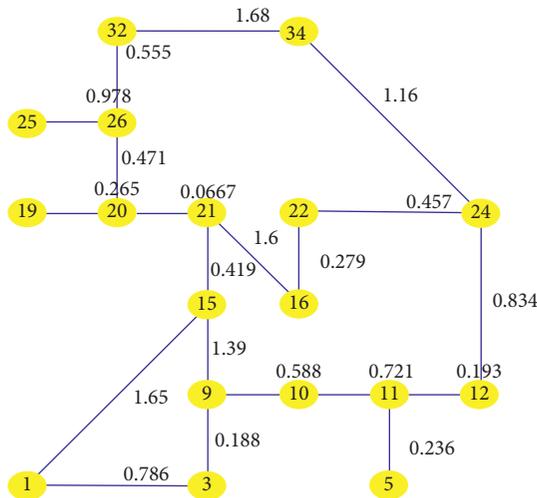


FIGURE 9: Diagram of the best possible timings obtained for the Iberian gauge railway network of SW Spain. Vertex 34 is Madrid, 19 is Badajoz, and 3 is Seville.

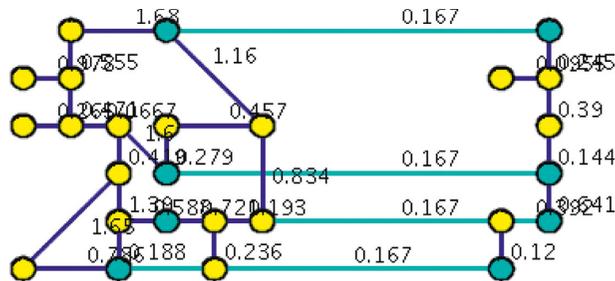


FIGURE 10: Diagram of graph  $G^*$  corresponding to SW Spain showing the best possible timings (in hours) for both Iberian and standard gauges subnetworks (and for passing through the gauge changeovers).

This way only the names of the stations and junctions along the route appear in the diagram, but it is possible to make other names to appear too. For instance,

```
PlottingRoute3d(H, Route_V_C,{"Sevilla_i",
"Sevilla_s", "Barcelona_i","Barcelona_s"});
PlotRoute3d;
```

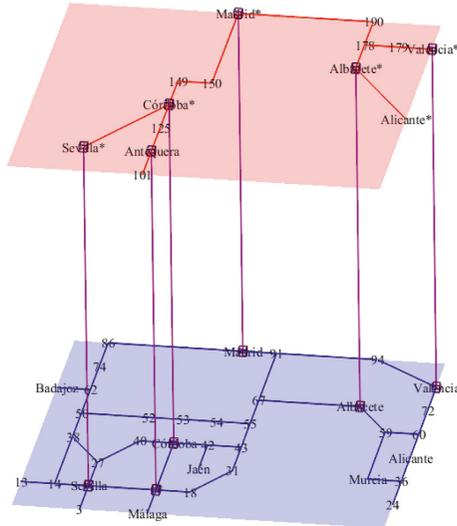


FIGURE 11: 3D plot of graph  $G^*$  corresponding to South Spain showing Iberian and standard gauge subnetworks and their connections (gauge changeovers).

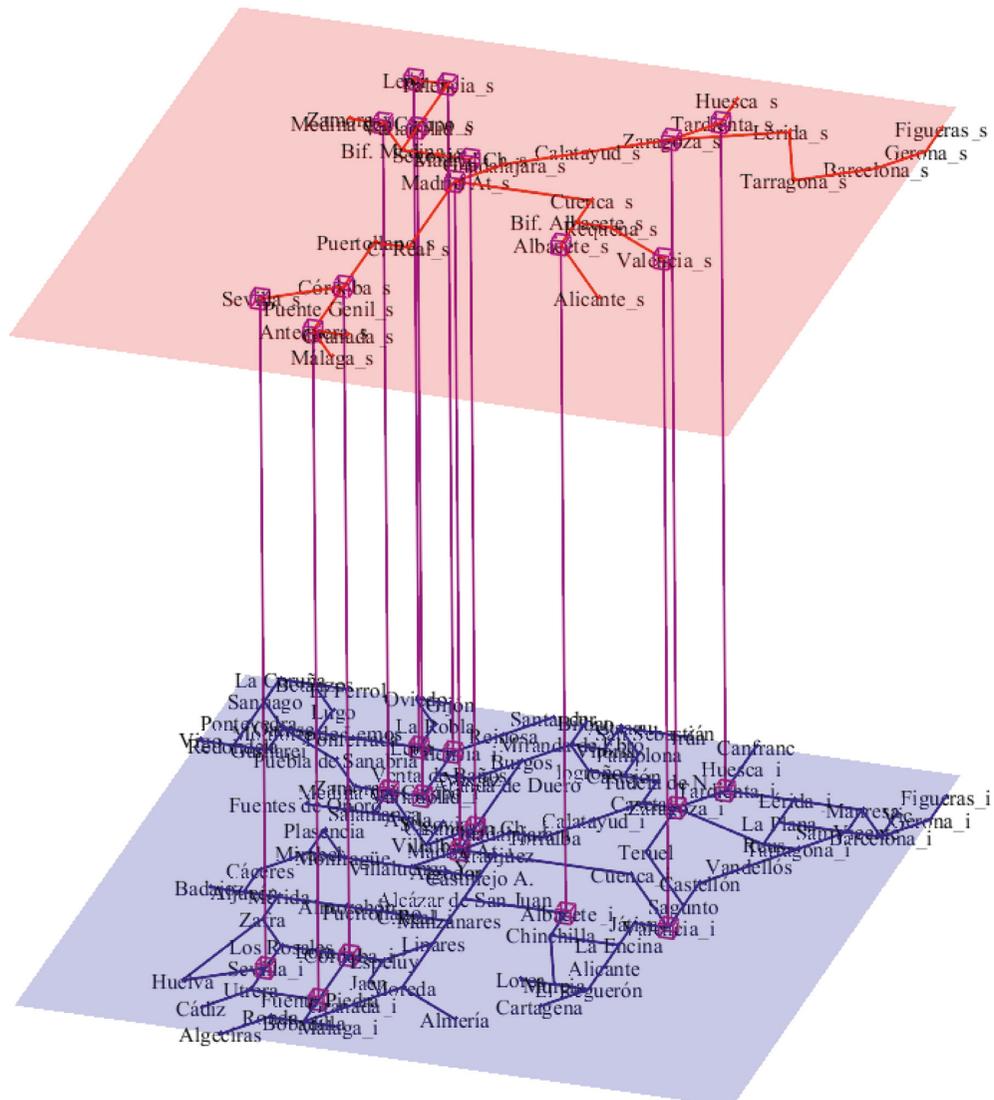


FIGURE 12: 3D plot of graph  $G^*$  corresponding to the whole Spanish railway network (excluding the narrow gauge subnetwork) showing the Iberian and standard gauges subnetworks (in the lower and upper planes, respectively) and the location of their connections (gauge changeovers).

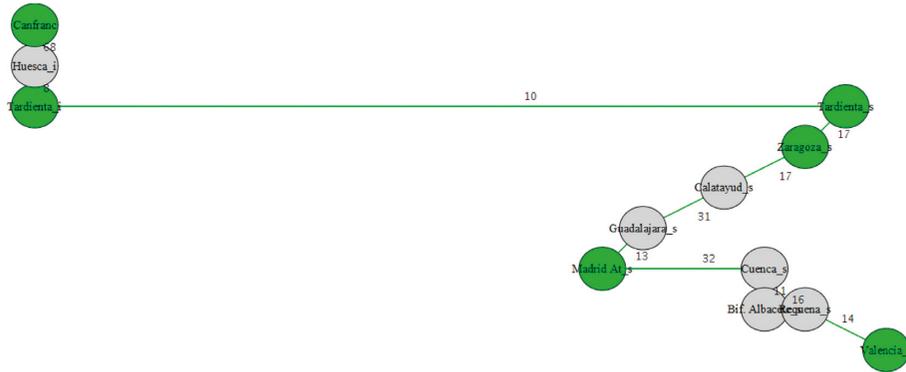


FIGURE 13: 2D plot of the Valencia-Canfranc best route (the part of the route where the train is set to standard gauge is on the right hand side, and the part where it is set to Iberian gauge is on the left hand side). The long horizontal connection represents the pass through a gauge changeover.

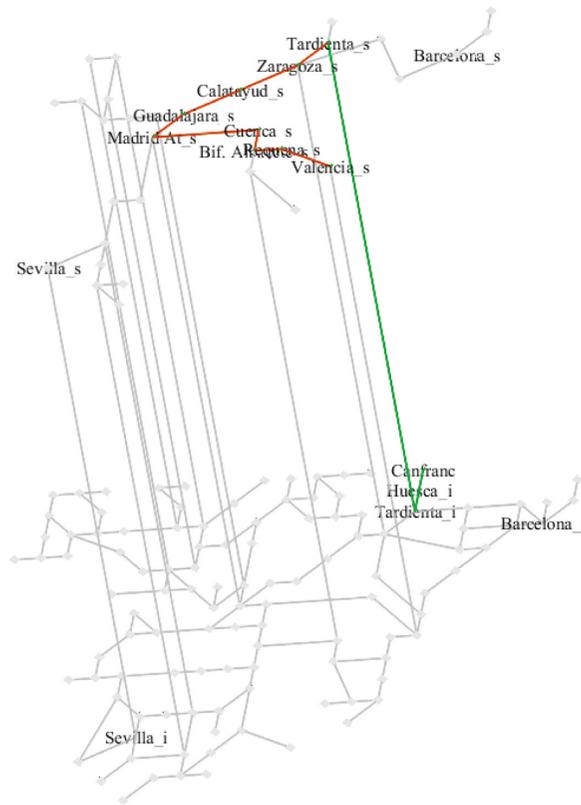


FIGURE 14: 3D plot of the Valencia-Canfranc best route. The part of the route where the train is set to standard gauge is on the upper plane (in red) and the part where it is set to Iberian gauge is on the lower plane (in green). The long vertical connection represents the pass through a gauge changeover (in green).

plot the same diagram but Sevilla\_i, Sevilla\_s, Barcelona\_i and Barcelona\_s, although not on the route, are correctly allocated in the corresponding plane of the 3D diagram (Figure 14).

### 5. Conclusions

Using an appropriate approach to shortest route finding and timings computation in a dual gauge railway network turns

it into a simple problem. This way any standard shortest route algorithm can be used, for instance, those included in the built-in graph packages of some computer algebra systems.

We know of no comparable package treating railway networks with dual gauge sections and gauge changeovers (what makes it impossible to directly use standard shortest route algorithms). The related package implemented by the

authors some years ago uses a completely different and far more complex mathematical model.

## Appendix

The data used in the Case Study of Section 4 can be found afterwards. The whole Adif network has been introduced (as said above, the narrow gauge subnetwork is excluded). Moreover, not all stations, but all main stations and all junctions, are introduced (in order to declare all sections of the network).

### (i) Iberian gauge:

```
introdato(nodo(12, 13), "Huesca_i", nodo(12, 14), "Canfranc", 135, 120):
```

```
introdato(nodo(7, 4), "Linares", nodo(6, 3), "Espelúy", 26, 135):
```

```
introdato(nodo(5, 3), "Córdoba_i", nodo(6, 3), "Espelúy", 101, 140):
```

```
introdato(nodo(5, 3), "Córdoba_i", nodo(4, 3), "Lora", 74, 160):
```

```
introdato(nodo(3, 3), "Los Rosales", nodo(4, 3), "Lora", 20, 160):
```

```
introdato(nodo(3, 3), "Los Rosales", nodo(2, 4), "Zafra", 40+99, (40+99)/(40*(1/120)+99/100)):
```

```
introdato(nodo(2, 5), "Mérida", nodo(2, 4), "Zafra", 65, 155):
```

```
introdato(nodo(2, 5), "Mérida", nodo(4, 5), "Almorchón", 121, 160):
```

```
introdato(nodo(5, 5), "Puertollano_i", nodo(4, 5), "Almorchón", 118, 135):
```

```
introdato(nodo(6, 5), "C.Real_i", nodo(5, 5), "Puertollano_i", 39, 140):
```

```
introdato(nodo(7, 5), "Manzanares", nodo(7, 4), "Linares", 42+67+9, (42+67+9)/(42*(1/160)+67/130+9/160)):
```

```
introdato(nodo(7, 5), "Manzanares", nodo(7, 6), "Alcázar de San Juan", 49, 160):
```

```
introdato(nodo(7, 7), "Castillejo A.", nodo(7, 6), "Alcázar de San Juan", 84, 160):
```

```
introdato(nodo(7, 7), "Castillejo A.", nodo(6, 7), "Algodor", 11, 140):
```

```
introdato(nodo(5, 7), "Villaluenga Y.", nodo(6, 7), "Algodor", 17, 80):
```

```
introdato(nodo(5, 7), "Villaluenga Y.", nodo(3, 7), "Monfragüe", 202, 155):
```

```
introdato(nodo(2, 7), "Mirabel", nodo(3, 7), "Monfragüe", 14, 155):
```

```
introdato(nodo(2, 7), "Mirabel", nodo(1, 6), "Cáceres", 66, 120):
```

```
introdato(nodo(1, 5), "Aljucén", nodo(1, 6), "Cáceres", 66, 120):
```

```
introdato(nodo(2, 5), "Mérida", nodo(1, 5), "Aljucén", 6, 100):
```

```
introdato(nodo(7, 7), "Castillejo A.", nodo(7, 8), "Aranjuez", 15, 160):
```

```
introdato(nodo(7, 8), "Aranjuez", nodo(6, 8), "Madrid At_i", 48, 160):
```

```
introdato(nodo(5, 7), "Villaluenga Y.", nodo(6, 8), "Madrid At_i", 32, 155):
```

```
introdato(nodo(2, 8), "Plasencia", nodo(3, 7), "Monfragüe", 17, 90):
```

```
introdato(nodo(2, 8), "Plasencia", nodo(2, 7), "Mirabel", 12, 80):
```

```
introdato(nodo(3, 3), "Los Rosales", nodo(3, 2), "Sevilla_i", 27, 160):
```

```
introdato(nodo(3, 1), "Bif. Utrera", nodo(3, 2), "Sevilla_i", 15, 160):
```

```
introdato(nodo(3, 1), "Bif. Utrera", nodo(5, 1), "Fuente Piedra", 70+41, (70+41)/(70*(1/160)+41/115)):
```

```
introdato(nodo(5, 3), "Córdoba_i", nodo(5, 1), "Fuente Piedra", 118, 155):
```

```
introdato(nodo(7, 4), "Linares", nodo(7, 2), "Moreda", 117, 140):
```

```
introdato(nodo(6, 1), "Granada_i", nodo(7, 2), "Moreda", 57, 140):
```

```
introdato(nodo(6, 1), "Granada_i", nodo(5, 0), "Bobadilla", 127, 155):
```

```
introdato(nodo(5, 1), "Fuente Piedra", nodo(5, 0), "Bobadilla", 11, 120):
```

```
introdato(nodo(7, 8), "Aranjuez", nodo(11, 8), "Cuenca_i", 152, 120):
```

```
introdato(nodo(11, 8), "Cuenca_i", nodo(13, 6), "Valencia_i", 199, 120):
```

```
introdato(nodo(12, 6), "Játiva", nodo(13, 6), "Valencia_i", 64, 160):
```

```
introdato(nodo(12, 6), "Játiva", nodo(12, 5), "La Encina", 48, 220):
```

```
introdato(nodo(10, 5), "Chinchilla", nodo(12, 5), "La Encina", 70, 160):
```

```
introdato(nodo(10, 5), "Chinchilla", nodo(10, 6), "Albacete_i", 20, 160):
```

```
introdato(nodo(7, 6), "Alcázar de San Juan", nodo(10, 6), "Albacete_i", 131, 200):
```

```
introdato(nodo(12, 4), "Alicante", nodo(12, 5), "La Encina", 79, 160):
```

```
introdato(nodo(12, 4), "Alicante", nodo(12, 3), "El Reguerón", 68, 160):
```

```
introdato(nodo(11, 3), "Murcia", nodo(12, 3), "El Reguerón", 10, 160):
```

```
introdato(nodo(11, 3), "Murcia", nodo
```

```

(10, 5), "Chinchilla", 157, 160):
introdada(nodo(3, 2), "Sevilla_i", nodo
(1, 1), "Huelva", 108, 140):
introdada(nodo(1, 1), "Huelva", nodo(2,
4), "Zafra", 51+83+47, (51+83+47)/(51*
(1/120)+83/85+47/120)):
introdada(nodo(1, 5), "Aljucén", nodo
(0, 5), "Badajoz", 53, 200):
introdada(nodo(12, 3), "El Reguerón",
nodo(11, 2), "Cartagena", 55, 160):
introdada(nodo(6, 0), "Málaga_i", nodo
(5, 0), "Bobadilla", 70, 160):
introdada(nodo(3, 1), "Utrera", nodo(2,
0), "Cádiz", 121+17, (121+17)/(17/140+
121/200)):
introdada(nodo(6, 2), "Jaén", nodo(6,
3), "Espelúy", 35, 140):
introdada(nodo(7, 2), "Moreda", nodo(9,
1), "Almería", 71+11+42, (71+11+42)/(71/
140+11/160+42*(1/100))):
introdada(nodo(5, 0), "Bobadilla", nodo
(4, 0), "Ronda", 70, 120):
introdada(nodo(3, -1), "Algeciras", no
do(4, 0), "Ronda", 106, 120):
introdada(nodo(11, 3), "Murcia", nodo
(10, 3), "Lorca", 157, 160):
introdada(nodo(5, 8), "Villalba", nodo
(6, 9), "Madrid Ch._i", 38, 160):
introdada(nodo(5, 8), "Villalba", nodo
(4, 9), "Avila", 83, 150):
introdada(nodo(3, 10), "M. del Campo_i",
nodo(4, 9), "Avila", 86, 160):
introdada(nodo(3, 10), "M. del Campo_i",
nodo(4, 10), "Valladolid_i", 42, 160):
introdada(nodo(4, 11), "Venta de
Baños", nodo(4, 10), "Valladolid_i",
37, 160):
introdada(nodo(4, 11), "Venta de
Baños", nodo(5, 11), "Magaz", 9, 160):
introdada(nodo(6, 12), "Burgos", nodo
(5, 11), "Magaz", 80, 160):
introdada(nodo(6, 12), "Burgos", nodo
(6, 11), "Aranda de Duero", 96, 100):
introdada(nodo(6, 9), "Chamartín Ch._i",
nodo(6, 11), "Aranda de Duero", 159, 130):
introdada(nodo(6, 9), "Chamartín Ch._i",
nodo(6, 8), "Madrid At_i", 10, 60):
introdada(nodo(5, 8), "Villalba", nodo
(5, 9), "Segovia_i", 43+20, (43+20)/
(43/100+2*(1/90))):
introdada(nodo(7, 9), "Guadalajara_i",
nodo(6, 8), "Madrid At_i", 57, 160):
introdada(nodo(7, 9), "Guadalajara_i",
nodo(8, 9), "Torralba", 99, 160):
introdada(nodo(9, 10), "Calatayud_i",
nodo(8, 9), "Torralba", 87, 160):
introdada(nodo(9, 10), "Calatayud_i",
nodo(10, 11), "Casetas", 36+47, (36+47)/
(36*(1/140)+47/160)):
introdada(nodo(11, 11), "Zaragoza_i",
nodo(10, 11), "Casetas", 12, 160):
introdada(nodo(11, 11), "Zaragoza_i",
nodo(11, 9), "Teruel", 177, 200):
introdada(nodo(13, 7), "Sagunto", nodo
(11, 9), "Teruel", 47+30+61, (47+30+61)/
(47/100+30*(1/140)+61/95)):
introdada(nodo(13, 7), "Sagunto", nodo
(13, 6), "Valencia_i", 33, 220):
introdada(nodo(6, 12), "Burgos", nodo
(7, 13), "Miranda de Ebro", 84, 160):
introdada(nodo(8, 12), "logroño", nodo
(7, 13), "Miranda de Ebro", 69, 110):
introdada(nodo(8, 12), "logroño", nodo
(9, 12), "Castejon", 76, 140):
introdada(nodo(10, 12), "Tudela de N.",
nodo(9, 12), "Castejón", 16, 160):
introdada(nodo(10, 12), "Tudela de N.",
nodo(10, 11), "Casetas", 62, 160):
introdada(nodo(7, 14), "Bilbao", nodo
(7, 13), "Miranda de Ebro", 63+40, 140):
introdada(nodo(8, 13), "Vitoria", nodo
(7, 13), "Miranda de Ebro", 33, 155):
introdada(nodo(8, 13), "Vitoria", nodo
(8, 14), "Altsazu", 43, 160):
introdada(nodo(9, 13), "Pamplona", nodo
(8, 14), "Altsazu", 52, 140):
introdada(nodo(9, 13), "Pamplona", nodo
(9, 12), "Castejón", 87, 140):
introdada(nodo(9, 14), "San Sebastián",
nodo(8, 14), "Altsazu", 87, 135):
introdada(nodo(9, 14), "San Sebastián",
nodo(10, 14), "Irún", 17, 135):
introdada(nodo(3, 9), "Salamanca", nodo
(4, 9), "Avila", 111, 155):
introdada(nodo(3, 9), "Salamanca", nodo
(3, 10), "Medina del Campo_i", 77, 155):
introdada(nodo(3, 9), "Salamanca", nodo
(1, 9), "Fuentes de Oñoro", 125, 140):
introdada(nodo(2, 10), "Zamora_i", nodo
(3, 10), "Medina del Campo_i", 90, 145):
introdada(nodo(2, 10), "Zamora_i", nodo
(1, 11), "Puebla de Sanabria", 107, 125):

```

```

introduata(nodo(0, 12), "Monforte de
Lemos", nodo(-1, 12), "Orense", 46, 155):
introduata(nodo(3, 12), "León_i", nodo(1,
12), "Ponferrada", 76+45, (76+45)/(76/
135+45*(1/155))):
introduata(nodo(3, 12), "León_i", nodo
(4, 12), "Palencia_i", 122, 160):
introduata(nodo(5, 13), "Reinosa", nodo
(4, 12), "Palencia_i", 79+50, (79+50)/
(79/160+50*(1/130))):
introduata(nodo(5, 13), "Reinosa", nodo
(6, 14), "Santander", 34+54, (34+54)/
(34/95+54*(1/160))):
introduata(nodo(3, 12), "León_i", nodo
(3, 13), "La Robla", 25, 110):
introduata(nodo(2, 14), "Oviedo", nodo
(3, 13), "La Robla", 31+83, (83+31)/(83/
105+31/130)):
introduata(nodo(2, 14), "Oviedo", nodo
(3, 14), "Gijón", 32, 140):
introduata(nodo(0, 12), "Monforte de
Lemos", nodo(0, 13), "Lugo", 71, 160):
introduata(nodo(-1, 14), "Betanzos", no
do(0, 13), "Lugo", 92, 160):
introduata(nodo(-1, 14), "Betanzos", no
do(-2, 14), "La Coruña", 26, 105):
introduata(nodo(2, 13), "Santiago", no
do(-2, 14), "La Coruña", 61, 200):
introduata(nodo(2, 13), "Santiago", no
do(-1, 12), "Orense", 88, 300):
introduata(nodo(-1, 14), "Betanzos", no
do(0, 14), "El Ferrol", 43, 90):
introduata(nodo(-1, 11), "Guillarei", no
do(-1, 12), "Orense", 95, 155):
introduata(nodo(-1, 11), "Guillarei",
nodo(2, 11), "Redondela", 26, 160):
introduata(nodo(2, 12), "Pontevedra",
nodo(2, 11), "Redondela", 18, 200):
introduata(nodo(2, 12), "Pontevedra",
nodo(2, 13), "Santiago", 68, 200):
introduata(nodo(-3, 11), "Vigo", nodo(2,
11), "Redondela", 18, 200):
introduata(nodo(13, 7), "Sagunto", nodo
(13, 8), "Castellón", 40, 200):
introduata(nodo(14, 9), "Vandellós", no
do(13, 8), "Castellón", 146, 220):
introduata(nodo(15, 10), "Tarragona_i",
nodo(14, 9), "Vandellós", 41, 160):
introduata(nodo(15, 10), "Tarragona_i",
nodo(14, 10), "Reus", 19, 140):

```

```

introduata(nodo(11, 11), "Zaragoza_i",
nodo(14, 10), "Reus", 182+48, (182+48)/
(182/135+48/155)):
introduata(nodo(11, 11), "Zaragoza_i",
nodo(12, 12), "Tardienta_i", 57, 160):
introduata(nodo(14, 12), "Lérida_i", no
do(12, 12), "Tardienta_i", 131, 160):
introduata(nodo(14, 11), "La Plana", no
do(14, 12), "Lérida_i", 69, 160):
introduata(nodo(14, 11), "La Plana", no
do(14, 10), "Reus", 21, 140):
introduata(nodo(14, 12), "Lérida_i", no
do(16, 12), "Manresa", 118, 155):
introduata(nodo(16, 11), "San Vicente",
nodo(16, 12), "Manresa", 53+26, (53+26)/
(53/140+26*(1/120))):
introduata(nodo(16, 11), "San Vicente",
nodo(14, 11), "La Plana", 36, 140):
introduata(nodo(16, 11), "San Vicente",
nodo(15, 10), "Tarragona_i", 25, 160):
introduata(nodo(16, 11), "San Vicente",
nodo(17, 11), "Barcelona_i", 60, 160):
introduata(nodo(17, 11), "Barcelona_i",
nodo(16, 12), "Manresa", 68, 140):
introduata(nodo(18, 12), "Gerona_i", no
do(17, 11), "Barcelona_i", 81, 140):
introduata(nodo(18, 13), "Figueras_i",
nodo(18, 12), "Gerona_i", 36, 140):
introduata(nodo(17, 12), "Vic", nodo(17,
11), "Barcelona_i", 74, 140):

```

(ii) Standard gauge (high speed lines):

```

t :=nodes:
introduata(t+nodo(6, 8), "Madrid At_s",
t+nodo(6, 5), "C. Real_s", 17+117, (17+
117)/(121/160+32*(1/300))):
introduata(t+nodo(6, 5), "C. Real_s", t+
nodo(5, 5), "Puertollano_s", 39, 270):
introduata(t+nodo(5, 5), "Puertollano_s",
t+nodo(5, 3), "Córdoba_s", 76+58, (76+58)/
(76*(1/270)+58*(1/250))):
introduata(t+nodo(5, 3), "Córdoba_s",
t+nodo(3, 2), "Sevilla_s", 127, 250):
introduata(t+nodo(5, 3), "Córdoba_s",
t+nodo(5, 2), "Puente Genil_s", 76, 300):
introduata(t+nodo(5, 2), "Puente Genil_s",
t+nodo(5, 1), "Antequera_s", 35, 300):
introduata(t+nodo(5, 1), "Antequera_s",
t+nodo(6, 0), "Málaga_s", 58, 300):
introduata(t+nodo(6, 8), "Madrid At_s",
t+nodo(10, 8), "Cuenca_s", 161, 300):

```

```

introdudata(t+nodo(10, 8), "Cuenca_s", t+
nodo(10, 7), "Bif. Albacete_s", 53, 300):
introdudata(t+nodo(10, 7), "Bif. Albacete
_s", t+nodo(10, 6), "Albacete_s", 73, 300):
introdudata(t+nodo(10, 6), "Albacete_s",
t+nodo(12, 4), "Alicante_s", 165, 300):
introdudata(t+nodo(10, 7), "Bif. Albace
te_s", t+nodo(11, 7), "Requena_s", 79, 300):
introdudata(t+nodo(11, 7), "Requena_s",
t+nodo(13, 6), "Valencia_s", 70, 300):
introdudata(t+nodo(6, 8), "Madrid At_s",
t+nodo(7, 9), "Guadalajara_s", 64, 300):
introdudata(t+nodo(9, 10), "Calatayud_s",
t+nodo(7, 9), "Guadalajara_s", 157, 300):
introdudata(t+nodo(9, 10), "Calatayud_s",
t+nodo(11, 11), "Zaragoza_s", 86, 300):
introdudata(t+nodo(14, 12), "Lérida_s",
t+nodo(11, 11), "Zaragoza_s", 141, 300):
introdudata(t+nodo(14, 12), "Lérida_s",
t+nodo(15, 10), "Tarragona_s", 81, 300):
introdudata(t+nodo(17, 11), "Barcelona_s",
t+nodo(15, 10), "Tarragona_s", 100, 300):
introdudata(t+nodo(17, 11), "Barcelona_s",
t+nodo(18, 12), "Gerona_s", 94, 255):
introdudata(t+nodo(18, 13), "Figueras_s",
t+nodo(18, 12), "Gerona_s", 34, 290):
introdudata(t+nodo(12, 12), "Tardienta_s",
t+nodo(11, 11), "Zaragoza_s", 86, 300):
introdudata(t+nodo(5, 1), "Antequera_s",
t+nodo(6, 1), "Granada_s", 86, 300):
introdudata(t+nodo(5, 9), "Segovia_s",
t+nodo(6, 9), "Madrid Ch_s", 67, 300):
introdudata(t+nodo(5, 9), "Segovia_s",
t+nodo(4, 9), "Bif. Medina_s", 66, 300):
introdudata(t+nodo(4, 10), "Valladolid_s",
t+nodo(4, 9), "Bif. Medina_s", 45, 300):
introdudata(t+nodo(4, 10), "Valladolid_s",
t+nodo(4, 12), "Palencia_s", 51, 300):
introdudata(t+nodo(3, 12), "León_s", t+
nodo(4, 12), "Palencia_s", 115, 300):
introdudata(t+nodo(3, 10), "Medina del Ca
mpo_s", t+nodo(4, 9), "Bif. Medina_s",
22, 200):
introdudata(t+nodo(3, 10), "Medina del
Campo_s", t+nodo(2, 10), "Zamora_s",
77, 200):

```

(iii) Dual gauge lines:

```

d:=2*nodes:
introdudata(d+nodo(12, 13), "Huesca", d+
nodo(12, 12), "Tardienta", 22, 160):
stations[nodo(12, 13)] := "Huesca_i":sta
tions[nodes+nodo(12, 13)] := "Huesca_s":

```

```

stations[nodo(12, 12)] := "Tardienta_i":sta
tions[nodes+nodo(12, 12)] := "Tardienta_s":
introdudata(d+nodo(18, 12), "Gerona", d+
nodo(18, 13), "Figueras", 34, 160):
stations[nodo(18, 12)] := "Gerona_i":sta
tions[nodes+nodo(18, 12)] := "Gerona_s":
stations[nodo(18, 13)] := "Figueras_i":sta
tions[nodes+nodo(18, 13)] := "Figueras_s":

```

## Data Availability

The data used in the case study of Section 4 can be found in the Appendix. The *Maple* code and worksheet of the mentioned case study are available from the corresponding author upon request.

## Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

## Acknowledgments

This work was partially supported by the research projects TIN2015-66471-P and PGC2018-096509-B-100 (Government of Spain) and grant CASI-CAM S2013/ICE-2845 (Comunidad de Madrid). The authors would like to thank the Fundación de los Ferrocarriles Españoles for providing data and maps of the Spanish railways.

## References

- [1] A. Hernando, E. Roanes-Lozano, A. García-Álvarez, L. Mesa, and I. González-Franco, "Optimal route finding and rolling stock selection in a dual gauge multi-voltage railway network," *Computing in Science & Engineering*, vol. 14, no. 4, pp. 82–89, 2012.
- [2] E. Roanes-Lozano, A. Hernando, A. García-Álvarez, L. Mesa, and I. González-Franco, "Calculating the exploitation costs of trains in the Spanish railways," *Computing in Science & Engineering*, vol. 15, no. 3, pp. 89–95, 2013.
- [3] E. Roanes-Lozano, A. Hernando, A. García-Álvarez, L. Mesa, and I. González-Franco, "Possibilities of RutasOptiRed package," in *XI Congreso de Ingeniería del Transporte (CIT 2014)*, A. Ibeas, J. L. Moura, L. dell'Olio, and B. Alonso, Eds., vol. 160, Elsevier Procedia-Social and Behavioral Sciences, Amsterdam, Netherlands, pp. 102–111, 2014.
- [4] A. Hernando, E. Roanes-Lozano, and A. García-Álvarez, "A recommender system for train routing: when concatenating two minimum length paths is not the minimum length path," *Applied Mathematics and Computation*, vol. 319, pp. 486–498, 2018.
- [5] E. Roanes-Lozano, A. Almech, C. Solano-Macías, and A. Hernando, "Determining the best routes on dual gauge railway networks using graphs," in *Proceedings of the 17th International Conference on Computational and Mathematical Methods in Science and Engineering CMMSE 2017*, J. Vigo-Aguiar, Ed., pp. 2219–2222, CMMSE, Cádiz, Spain, 2017.

- [6] Dirección General de Negocios y Operaciones Comerciales. Dirección de Gabinete y Gestión Corporativa. Adif, *Declaración sobre la red 2019. V.0*, Adif, Madrid, Spain, 2019.
- [7] Adif-Fichas Informativas, *Tercer carril*, 2019, [http://www.adif.es/es\\_ES/ocio\\_y\\_cultura/fichas\\_informativas/ficha\\_informativa\\_00042.shtml](http://www.adif.es/es_ES/ocio_y_cultura/fichas_informativas/ficha_informativa_00042.shtml).
- [8] Dirección General de Negocio, Operaciones Comerciales. Dirección de Gabinete y Gestión Corporativa, Adif, *Declaración sobre la red 2019. Mapas de la Red titularidad de Adif 2019. V.0*, Adif, Madrid, Spain, 2019.
- [9] Adif-Fichas Informativas, *Cambiadores automáticos de ancho de vía*, 2019, [http://www.adif.es/es\\_ES/ocio\\_y\\_cultura/fichas\\_informativas/ficha\\_informativa\\_00034.shtml](http://www.adif.es/es_ES/ocio_y_cultura/fichas_informativas/ficha_informativa_00034.shtml).
- [10] A. García-Álvarez, *Automatic Track Gauge Changeover for Trains in Spain*, Fundación de los Ferrocarriles Españoles, Madrid, Spain, 2010.
- [11] 2019 [http://www.renfe.com/viajeros/nuestros\\_trenes/index.html](http://www.renfe.com/viajeros/nuestros_trenes/index.html).
- [12] Y. Gao, L. Yang, and S. Li, "Uncertain models on railway transportation planning problem," *Applied Mathematical Modelling*, vol. 40, no. 7-8, pp. 4921-4934, 2016.
- [13] E. Hassannayebi and S. H. Zegordi, "Variable and adaptive neighbourhood search algorithms for rail rapid transit timetabling problem," *Computers & Operations Research*, vol. 78, pp. 439-453, 2017.
- [14] L. Kang and X. Zhu, "Strategic timetable scheduling for last trains in urban railway transit networks," *Applied Mathematics and Computation*, vol. 45, pp. 209-225, 2017.
- [15] L. F. Escudero and S. Muñoz, "A survey-based approach for designing the lines of a rapid transit network," *Discrete Applied Mathematics*, vol. 210, pp. 14-34, 2016.
- [16] J. L. Galán-García, G. Aguilera-Venegas, and P. Rodríguez-Cielos, "An accelerated-time simulation for traffic flow in a smart city," *Journal of Computational and Applied Mathematics*, vol. 270, pp. 557-563, 2014.
- [17] J. L. Galán-García, G. Aguilera-Venegas, M. Á. Galán-García, and P. Rodríguez-Cielos, "A new Probabilistic Extension of Dijkstra's Algorithm to simulate more realistic traffic flow in a smart city," *Applied Mathematics and Computation*, vol. 267, pp. 780-789, 2015.
- [18] F. Dou, K. Yan, Y. Huang, L. Wang, and L. Jia, "Optimal path choice in railway passenger travel network based on residual train capacity," *The Scientific World Journal*, vol. 2014, p. 153949, 2014.
- [19] E. Ortega Pérez, S. Mancebo Quintana, and I. Otero Pastor, "Road and railway accessibility atlas of Spain," *Journal of Maps*, vol. 7, no. 1, pp. 31-41, 2011.
- [20] F. Schulz, D. Wagner, and K. Weihe, "Dijkstra's algorithm on-line: an empirical case study from public railroad transport," in *WAE'99, LNCS 1668*, J. S. Vitter and C. D. Zaroliagis, Eds., pp. 110-123, Springer-Verlag, Berlin, Germany, 1999.
- [21] F. Schulz, D. Wagner, and K. Weihe, "Dijkstra's algorithm on-line," *Journal of Experimental Algorithmics*, vol. 5, p. 12, 2000.
- [22] S. Pallottino and M. G. Scutellà, "Shortest path algorithms in transportation models: classical and innovative aspects," in *Equilibrium and Advanced Transportation Modelling*, Centre for Research on Transportation, P. Marcotte and S. Nguyen, Eds., Springer US, Berlin, Germany, pp. 245-281, 1998.
- [23] E. W. Dijkstra, "A note on two problems in connexion with graphs," *Numerische Mathematik*, vol. 1, no. 1, pp. 269-271, 1959.
- [24] R. W. Floyd, "Algorithm 97: shortest path," *Communications of the ACM*, vol. 5, no. 6, p. 345, 1962.
- [25] Adif Dirección Ejecutiva de Circulación, *CIRCulaciones por TRAMos (CIRTRA) 2014*, Adif, Madrid, Spain, 2015.
- [26] J. Ribeiro da Silva and M. Ribeiro, *Os Comboios em Portugal*, Terramar, Lisboa, Portugal, 2009.
- [27] E. Roanes-Lozano, J. Gómez-Castaño, and C. Solano-Macías, "Evolution of the connectivity of the Portuguese broad gauge railway network (1948-2012)," in *Railways: Types, Design and Safety Issues*, C. Reinhardt and K. Shroeder, Eds., pp. 149-164, Nova Science Publishers, Inc., New York, NY, USA, 2013.
- [28] E. Roanes-Lozano, L. M. Laita, E. Roanes-Macías, M. J. Wester, J. L. Ruiz-Lozano, and C. Roncero, "Evolution of railway network flexibility: the Spanish broad gauge case," *Mathematics and Computers in Simulation*, vol. 79, no. 8, pp. 2317-2332, 2009.

