Time control in road passenger transportation. Problem description and formalization*

Beatriz Lopez

Institut d'Informàtica i Aplicacions, Universitat de Girona,
Campus Montilivi, edifice P4, 17071 Girona, Spain

Abstract. The aim of this report is to provide a description and formalization of the road passenger transportation problem to be faced in project TIN2004-06354-C02-02. This document is the basis for future development of techniques and methods that eventually provide some solution to the problem.

* IIIA Research Report 05-04, University of Girona

Table of Contents

1 Introduction

Road passenger transportation has been for years a matter of concern for the traffic responsible in order to minimize bus accidents. Traffic accidents in general are one of the major mortality rates in developed countries. In this line, several European governments are campaigning for better driving practices. Regarding buses, the European law is also evolving in order to control professional driving licences and driving times, with the aim of assuring the maximum guarantees to the citizen that use road passenger transports.

This new laws and regulations are posing a lot of requirements to the companies related to this economic field. The challenge is not so much related to regular and down town services that can be scheduled once a year, but to just-in-time services. That is, services required within a short period of time, usually, from one day to the next one. This kind of services are often related to conference events, holidays, excursions, etc. This kind of services are provided by inter-urban transport companies.

In the past, there was a human operator in the inter-urban transport companies in charge of allocating once a day drivers to required services. For example, at night, when all the customers have already performed they requests, the operator dedicate so much time to the allocation process. New laws and legislation, however, are posing too many constraints for a human operator. As a bypass solution, operators elaborate schedulers in which drivers have unoccupied hours. The economic consequences for the companies benefits are evident: with the same amount of drivers, they can provide less services, and so they earn less money. Moreover, there is no guarantee that all the constraints imposed by the law are satisfied, so the company is assuming the risk to be billed by the traffic authorities.

Trying to face this problem, we have proposed to the Spanish Ministry of Education a research project for finding new scheduling techniques that solve this problem. The aim of this report is to formally describe the road transportation problem, so that is serves as a working document for further developments of scheduling techniques.

2 Problem description

Given:

- Resources:
  - Drivers: $D = \{d_1, \ldots, d_n\}$
  - Buses: $B = \{b_1, \ldots, b_m\}$
- Tasks:
  - Services: $S = \{s_1, \ldots, s_l\}$

assign to each service a driver and a bus, subject to the constraints and preferences provided below.

The problem can be split into two main parts:

- Driver allocation
- Bus allocation

There are two main reasons for that. First, each driver has a bus assigned by default, so the second allocation process is trivial when solving the first one. And second, in case that additional buses were required, there is no problem to rent extra ones. The critical resources are drivers.

The time unit of measurement is hours.

The account of all the times is performed for a *sliding time window* of one month. For convenience, we consider a month composed by 28 days organized in four weeks:

- week 1: 1, ..., 7
- week 2: 8, ..., 14
- week 3: 15, ..., 21
- week 4: 22, ..., 28

All the time definitions that follows and constraints defined along the report should be contextualized within this sliding time window.

3 Services

Definition 1. A *service* is a tuple

$$s_i =< ti_i, tf_i, dur_i, orig_i, dest_i, n_i, D^i, b_i >$$

where

- ti_i: initial time
- tf_i: final time, $tf_i > ti_i$
- dur_i: duration
- $orig_i$: origin
- $dest_i$: destination
- n_i: number of passengers
- D^i: drivers assigned, $D^i = \{d^i_1, \ldots, d^i_{p_i}\}$ and $|D^i| \geq 1$
- bi: bus assigned

Note that the set of drivers D^i can be sorted according to the time slot assigned to drivers along the service duration. This precedence relationship is pointed out in the problem formulation (see section 8).

Further refinements:

1. Include itineraries: $Iti =< track_{i_1}, \ldots track_{i_k} >$, where $track_{i_1} = orig_i$ and $track_{i_k} = dest_i$

4 Buses

Definition 2. A *bus* is a tuple

$$bi =< n_i, p_i, pkm_i >$$

where:

- n_i: number of passengers (= capacity)
- p_i: basic cost
- pkm_i: cost per kilometer

5 Drivers

Definition 3. A *driver* is a tuple

$$d_i =< \mathcal{T}_i^d, \mathcal{T}_i^p, \mathcal{T}_i^b, \mathcal{T}_i^w, p_i, pkm_i >$$

where $\mathcal{T}_i^d, \mathcal{T}_i^p, \mathcal{T}_i^b$, and $\mathcal{T}_i^w$ are four different time measures, namely, effective working time, presence time, break time and weekly-break time, p_i is the basic cost and pkm_i is the cost per kilometer.

5.1 Effective working time

The *effective working time* $\mathcal{T}_i^d$ measures the time the driver i is effectively driving a bus. This time includes auxiliary works.

Definition 4. The *effective working time* $\mathcal{T}_i^d$ for driver i is defined as the set of all dairy effective working times within the sliding time window:

$$\mathcal{T}_i^d = \{T_i^{d_1}, \ldots, T_i^{d_{28}}\}$$

where: $T_i^{d_j}$ is the dairy effective working time for day j.

Definition 5. The *dairy effective working time for day j*, $T_i^{d_j}$, is the sequence of all time slots assigned to driver i for driving a bus along journey j:

$$T_i^{d_j} = t_{i_{j_1}}^{d_j} < \ldots < t_{i_{j_d}}^{d_j}$$

Each time slot $t_{i_k}^{d_j}$ represents the initial time in which the driver should start a given service. The duration of the time slot $t_{i_k}^{d_j}$ is noted as $|t_{i_k}^{d_j}|$. Note that $|t_{i_k}^{d_j}|$ cannot necessarily equals the duration of the service. According to the different constraints, several drivers can be assigned to a service, so a time slot can partially cover the service duration. It is the addition of the time slots of all the drivers assigned to a service that should totally cover the service duration (see problem formulation at section 8).

Definition 6. The accumulated effective working time for day j is defined as:

$$T_i^{\Sigma d_j} = \sum_{k=i_{j_1}}^{i_{j_d}} |t_k^{d_j}|$$

5.2 Presence time

The *presence time* $\mathcal{T}_i^p$ measures the time the driver is in the bus but not driving.

Definition 7. The *presence time* $\mathcal{T}_i^p$ for driver i is defined as the set of all dairy presence times within the sliding time window:

$$\mathcal{T}_i^p = \{T_i^{p_1}, \ldots, T_i^{p_{28}}\}$$

where $T_i^{p_j}$ is the dairy presence time in day j.

Definition 8. The *dairy presence time for day j*, $T_i^{p_j}$, is the sequence of all time slots assigned to driver i along journey j in which he/she is not driving:

$$T_i^{p_j} = t_{i_{j_1}}^{p_j} < \ldots < t_{i_{j_p}}^{p_j}$$

There should be a relationship between two consecutive effective working time slots, t_i^d and t_{i+1}^d and a presence time slot in between, t_j^p. That is, if two consecutive effective time slots have some time gap, such time gap should correspond to a presence time slot. Formally:

Theorem 9. *If* $t_i^d + |t_i^d| < t_{i+1}^d$*, then* $\exists t_j^p$ *such that* $t_i^d + |t_i^d| < t_j^p + |t_j^p| < t_{i+1}^d$.

This relationship, as the presence time is explicitly represented, should not be necessary to take into account when finding solutions to the problem.

Definition 10. The accumulated presence time for day j is defined as:

$$T_i^{\Sigma p_j} = \sum_{k=i_{j_1}}^{i_{j_p}} |t_k^{p_j}|$$

5.3 Break time

The *break time* $\mathcal{T}_i^b$ measures the time the driver is out of the vehicle along its journey. The minimum length is one hour.

Definition 11. The *break time* $\mathcal{T}_i^b$ for driver i is defined as the set of all dairy break times within the sliding time window:

$$\mathcal{T}_i^b = \{T_i^{b_1}, \ldots, T_i^{b_{28}}\}$$

where $T_i^{b_j}$ is the dairy break time for day j.

Definition 12. The *dairy break time for day j*, $T_i^{b_j}$, is the sequence of all time slots assigned to driver i along journey j in which he/she is out of the car:

$$T_i^{b_j} = t_{i_{j_1}}^{b_j} < \ldots < t_{i_{j_b}}^{b_j}$$

where each $|t_{i_{j_k}}^{b_j}| \geq 1$.

Definition 13. The accumulated break time for day j is defined as:

$$T_i^{\Sigma b_j} = \sum_{k=i_{j_1}}^{i_{j_b}} |t_k^{b_j}|$$

5.4 Weekly break time

The *weekly break time* $\mathcal{T}_i^w$ measures the time the driver has continuous break along a week (week ends, holiday). Weekly break time includes dairy break time. Both concepts, break and weekly-break should be considered as separated entities related to constraints required by the UE.

Definition 14. The *weekly break time* $\mathcal{T}_i^b$ for driver i is defined as the set of all four break times corresponding to the four weeks within the sliding time window:

$$\mathcal{T}_i^w = \{T_i^{w_1}, \ldots, T_i^{w_4}\}$$

where $T_i^{w_j}$ weekly break time in week j

Definition 15. The *weekly break time for week j*, $T_i^{w_j}$, is the sequence of all time slots assigned to driver i along week j in which he/she is either out of the office:

$$T_i^{w_j} = t_{i_{j_1}}^{w_j}, \ldots, t_{i_{j_w}}^{w_j}$$

Definition 16. The accumulated week break time for week j is defined as:

$$T_i^{\Sigma w_j} = \sum_{k=i_{j_1}}^{i_{j_w}} |t_k^{w_j}|$$

Note that a week models 7 continuous days within the sliding time window :

- week 1: 1, ..., 7
- week 2: 8, ..., 14
- week 3: 15, ..., 21
- week 4: 22, ..., 28

In this sense there is not distinction among Monday, Tuesday, Wednesday, ..., Sunday.

The break time is included in the weekly time, and such relationship is formalized according the following equations:

$$t_{i_{j_1}}^{w_j} = \sum_{k=1}^{7} T_i^{\Sigma b_k}, t_{i_{j_1}}^{w_j} = \sum_{k=8}^{14} T_i^{\Sigma b_k}, t_{i_{j_1}}^{w_j} = \sum_{k=15}^{21} T_i^{\Sigma b_k}, t_{i_{j_1}}^{w_j} = \sum_{k=22}^{28} T_i^{\Sigma b_k} \qquad (1)$$

6 Constraints

6.1 Coverage

The addition of all the time slots of effective working time of the drivers allocated to a service, should cover the duration of the service. That is:

$$\forall s_i \in S, \sum_k |t^x_{k_j}| = dur_i \tag{2}$$

where $t^x_{k_j} \in T^{d_x}_k \in d^i_k \in D^i$.

6.2 Overlapping constraints

o.1 Different services with common drivers assigned should have non overlapping times.

$$s_i \neq s_j, \forall d_k \in D^i \cap D^j \rightarrow tf_i < ti_j \vee tf_j < ti_i$$

o.2 Different services with common buses assigned should have non overlapping times.

$$s_i \neq s_j, b_i = b_j \rightarrow tf_i < ti_j \vee tf_j < ti_i$$

When including itineraries, these constraints should be revised to add time gaps between non consecutive services at different places.

6.3 Constraints on effective working time

w.1 Maximum within one day: 12h

$$\forall i, j, T_i^{\Sigma d_j} + T_i^{\Sigma p_j} \leq 12$$

w.2 Maximum driving time: 9h

$$\forall i, j, T_i^{\Sigma d_j} \leq 9$$

- Exception: 10h twice a week.

$$\forall i, j, (T_i^{\Sigma d_j} \leq 9) \vee (9 < T_i^{\Sigma d_j} \leq 10 \wedge ......)$$

w.3 Maximum driving time in two weeks: 90h
- Driving time in week 1: $T_i^{D_1} = \sum_{k=1}^{7} T_i^{\Sigma d_k}$
- Driving time in week 2: $T_i^{D_2} = \sum_{k=8}^{14} T_i^{\Sigma d_k}$
- Driving time in week 3: $T_i^{D_3} = \sum_{k=15}^{21} T_i^{\Sigma d_k}$
- Driving time in week 4: $T_i^{D_4} = \sum_{k=22}^{28} T_i^{\Sigma d_k}$

$$\forall i, k, T_i^{D_k} + T_i^{D_{k+1}} \leq 90$$

w.4 Maximum continuous driving time:
 – Urban transport: 6h – It is not the case
 – Minor transport: 2h It is not the case (?)
 – Otherwise: 4h30

$$\forall i, j, k, t_{i_k}^{d_j} \leq 4.5$$

Violation of any constraint up to 20% of the time, is considered a minor fault.

6.4 Constraints on presence time

p.1 Maximum: 20h per week in average (in a month period)
 – Presence time in average for week 1: $T_i^{P_1} = \frac{\sum_{k=1}^{7} T_i^{\Sigma p_k}}{7}$
 – Presence time in average for week 2: $T_i^{P_2} = \frac{\sum_{k=8}^{14} T_i^{\Sigma p_k}}{7}$
 – Presence time in average for week 3: $T_i^{P_3} = \frac{\sum_{k=15}^{21} T_i^{\Sigma p_k}}{7}$
 – Presence time in average for week 4: $T_i^{P_4} = \frac{\sum_{k=22}^{28} T_i^{\Sigma p_k}}{7}$

$$\forall i, k, T^{P_k} \leq 20$$

This time do not compute neither in the effective journey nor the extra months pay.

6.5 Constraints on break time

b.1 Minimum continuous break time between two consecutive journeys: 11h

$$\forall i, j, t_{i_{j_b}}^{b_j} + t_{i_1}^{b_{j+1}} \geq 11$$

 – Exception: 9h three times a week
  • Consequences: This time should be compensated throughout the remaining days of the week (and before the end of the week).
 – Exception: In case of work shift (from morning to afternoon and vice versa): 7h
  • Consequences: compensation of 5 hours.

b.2 In case that the time is split in several bits:
 – At least one of the bits is 8h long
 – The remaining bits are at least 1h long.
 – The total amount of all the bits is 12h.

$$|T_i^{b_j}| > 1 \rightarrow (\exists k, t_{i_k}^{b_j} \geq 8) \wedge (\forall q \neq k, t_{i_q}^{b_j} \geq 1) \wedge (T_i^{\Sigma b_j} \geq 12)$$

b.3 Vehicles with two drivers: minimum continuous break time 8h within 30hours.

$$if \exists s_i, |D^i| > 1, \ldots.$$

NOTE: (b.1 or b.2)

6.6 Constraints on weekly break time

b.4 Minimum continuous weekly break time:
  - At home: 36 hours
  - Out of home: 24 hours
  - Normal/recommended: 45 hours.

  If this time is less than 45 hours, the differential should be recovered in the next three weeks.

b.5 Discretionary transport: two weakly break periods after 12 days.

7 Preferences

7.1 Bus preferences

Buses with low cost are preferred than expensive buses.

7.2 Driver preferences

Cost. Drivers with low cost are preferred than expensive drivers. Low cost drivers means 0 basic cost, since they are employer of the company. Otherwise, drivers are hired as required.
Continuity Continuity means any of the following situations:

- Few slots of working time
- Services assigned to two continuous time slots, are close in the time

$$\forall k, t_{i_k}^{d_j}, t_{i_{k+1}}^{d_j} \in d_j, |tf_k - ti_{k+1}| < \varepsilon$$

8 Problem formulation

Definition 17. *Driver's allocation problem.* Given a set of services $S = \{s_1, ..., s_l\}$ required in day x, and a set of drivers D, assign a set of drivers $D^i \in 2^D$ for each service s_i subject to the constraints of coverage, o.1-o.2, w.1-w.4, p.1, b.1-b.5, and preferences on cost and continuity.

Each driver $d_k^i \in D^i$ would then have allocated at leat one time slot in his effective working time $t_{k_j}^x \in T_k^{d_x}$ for service i, and eventually some time slot in his presence time $t_{k_y}^x \in T_k^{p_x}$. The remaining time of the day is allocated as break and weekly-break time for the corresponding driver.

Different solutions are feasible. Each solution sol has a global cost $c(sol)$ that relates the number of constraints violated, the number of preferences unsatisfied, the drivers cost and the buses cost. Defining an appropriate cost function will be a matter of study for future work. The optimization problem consists then on finding the best solution. Formally:

Definition 18. *Driver's optimization problem.* Given a set of services $S = \{s_1, ..., s_l\}$ required in day x, a set of drivers D, and a function cost c, find the set of drivers $D^i \in 2^D$ for each service s_i subject to the constraints of coverage, o.1-o.2, w.1-w.4, p.1, b.1-b.5, and preferences on cost and continuity so that c is maximized.

9 Computational complexity

For every service at leat one driver should be assigned. Depending on the service duration, one service may require one, two, three different drivers. In general, a service can be assigned with a subset of drivers in 2^D, being D the current set of drivers, where $|D| = n$ (see drivers definition) . There are l services, so a first approach could point out a total amount of $|2^D|^l$ combinations. The the complexity of the problem growths exponentially to the number of services requested.

This complexity cost can be slightly lowered with the following considerations. If only one driver should be assigned, then only n^l combinations should be analyzed, where $|2^D|^l >> n^l$ (as $n < |2^D|$) . One can assume, then, that in case of assigning more than one driver to a service, probably two drivers would be assigned, but a bigger amount is unfeasible. So, we can simplify and assume that the number of combinations is $o(n^l)$.

10 Conclusions

In this paper we have described and formalized the problem of road passenger transportation. As it is shown, we are dealing with a very difficult optimization problem. Even the modelisation of the problem provided in this paper has required a lot of effort.

Future work goes towards the development of scheduling techniques that contribute to the solution of the problem. The ultimate goal is to have an automate tool that can provide schedulers anytime, so it was possible to deal with just-in-time requests and any kind of incidences (bus break down, traffic jump, etc.) .

Acknowledgements

This work was supported by the Spanish MEC Project TIN2004-06354-C02-02.

Appendix A. Notation

A.1 Constants

n	Number of drivers
m	Number of buses
l	Number of services
k	Number of tracks in an itinerary
p_i	Number of drivers assigned to service s_i
i_{j_d}	Number of time slots for effective working time for day j of driver i
i_{j_p}	Number of time slots for presence time for day j of driver i
i_{j_b}	Number of time slots for dairy break time for day j of driver i
i_{j_w}	Number of time slots for weekly break time for week j of driver i

A.1 Variables

d	drivers
s	services
b	buses
t	Inicial time
tf	Final time
dur	duration
orig	Origen
dest	Destination
n	Number of passengers
it	Itinerary
track	Track
p	Basic prize (buses)
pkm	Prize per kilometer (buses)
$\mathcal{T}^d$	Effecive working time
$\mathcal{T}^p$	Presence time
$\mathcal{T}^b$	Dairy break time
$\mathcal{T}^w$	Weekly break time

This article was processed using the LaTeX macro package with LLNCS style