An Exact Label Setting Algorithm for the Truck Driver Scheduling Problem considering European Community Social Legislation

Thierry Garaix, Philippe Lacomme, Nicolay Tchernev, Iván Peña-Arenas

An Exact Label Setting Algorithm for the Truck Driver Scheduling Problem considering European Community Social Legislation

Thierry Garaix

CNRS, LIMOS UMR 6158

Ecole des Mines de Saint-Etienne, Saint-Etienne, France

Philippe Lacomme

CNRS, LIMOS UMR 6158

Université Clermont Auvergne, Aubière, France

Nicolay Tchernev

CNRS, LIMOS UMR 6158

Université Clermont Auvergne, Aubière, France

Iván Peña-Arenas

CNRS, LIMOS UMR 6158

Université Clermont Auvergne, Aubière, France

Email: ivan_guillermo.pena_arenas@uca.fr

1 Introduction

Different works have been achieved for the Truck Drivers Scheduling Problem (TDSP), both, independently or as a subproblem in a vehicle routing procedure. Most of them consider a planning horizon of one week, incompletely include the set of rules of the EC social legislation and try to compute feasible solutions [1][2]. Kok et al. [3] have introduced a dynamic programming heuristic considering all the rules, albeit as in previous approaches, they make a simplification on working time breaks during a shift and do not include the night working rule. Using a planning horizon of two weeks and considering the Regulation (EC) No 561/2006 on driving hours, Drexl and Prescott-Gagnon [4] describe exact and heuristic labelling setting algorithms to find legal vehicle routes and schedules. Additionally they propose the conjecture of NP-completeness for this problem. Goel [5], and Tilk and Goel [6] find a feasible solution for the TDSP over a period of one week, considering all the set of rules, albeit, simplifying the rule on working time breaks during a shift and forbidding the night work.

The night working constraint states: if night work is performed, the daily working time does not exceed ten hours in each 24h period; from the algorithmic point of view, this entails two challenges, updating the resources in a sliding 24h time window and designing efficient dominance rules to achieve optimality in a competitive computational time.

This paper presents an exact label setting algorithm that takes into account the set of rules from the European Community social legislation, (EC) No 561/2006 on driving hours and Directive 2002/15/EC on working hours, over a period of one week. Moreover, no exact nor heuristic solution method has been proposed to solve the TDSP, which addresses all restrictions from the EC social legislation. In addition, a MILP model is used to compare the performance and quality of the results. Section 2, briefly describes the algorithm and shows comparisons, and conclusions are presented in section 3.

2 Label setting algorithm

2.1 Algorithm description

The method considers a sequence σ of customers that starts and ends at the depot, assuming that each of them has time windows and two activities: service and driving, where a given activity could have a duration of zero units of time. The objective is to find the optimal schedule of breaks/rests complying with the European Community social legislation, which minimizes the completion time of the sequence.

The general assumption is that exists δ such that each break duration, time windows, service and traveling time are multiple of δ. Extending with idle time is only possible at the beginning of the sequence or at the beginning of the shift. If $\delta \geq 0.25h$, idle is only dominant at the beginning of the sequence or at the beginning of the shift, otherwise a break is dominant. In general, if $\delta < 0.25h$ then extensions with idle would be useful at all times, since, them could be used to avoid rules infractions.

Let A be the set of activities and BD the set of all types of breaks $BD = \{0h, 0.25h, 0.50h, 0.75h, 3h, 9h, 11h\}$. The algorithm starts with the service activity at the depot, using an initial solution set to 0 ($\lambda_0 \leftarrow 0$). At each activity $i \in A$, it determines a set of feasible solutions or labels (S, i) with a minimum completion time. There are two types of extensions for a label $\lambda_k \in (S, i)$. First, process p (service/driving) followed by a break b, in this order. The size of the process p is the maximum process time until a rule could be broken, an activity is finished or it is the beginning of the night interval. Second, idle followed by a break. It is done, in order to postpone the beginning of the shift or if the current time is before the earliest starting time of the next customer of the sequence. In both cases, all types of breaks from set BD are investigated. After each extension, the label updates their attributes/resources according to the type of activity, its duration and the type of break. When using a planning horizon of one week most of the rules apply during a shift, thus the resources related to them reset their values to zero when a shift ends, that is to say a rest/break of 9h or more, takes place. One of the inconveniences with the night rule is updating their attributes, because unlike the other resources they do not reset their values to zero at the end of a given shift. Figure 1 depicts the night working rule.

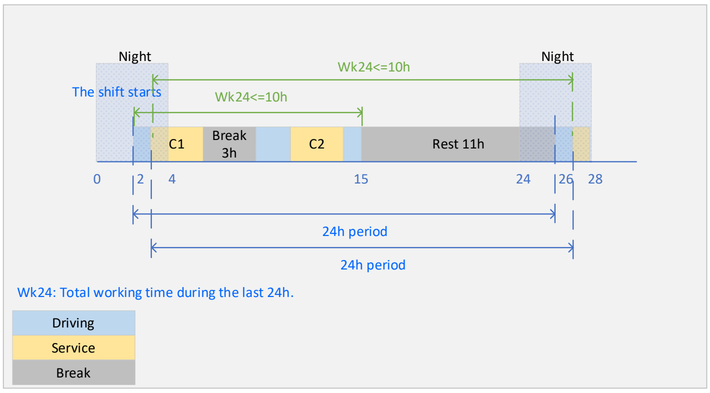

Figure 1. Night work rule.

Therefore, updating the attributes related with this rule, i.e. work performed during the night or not and the total working time during the last 24h, implies to keep track of their information during the last 24h at each extension, instead of resetting them to zero at the end of each shift.

In general, during the dominance procedure if the used resources of label $l1$ are less or equal to those of label $l2$, then $l1$ could dominate $l2$. In the night case, the last 24h period of a label is decomposed in sub-intervals of size δ, and for each of them the cumulated working time is computed. The comparison is done for each sub-interval and if the cumulated working time of a label $l1$ is less or equal to the cumulated working time of $l2$, then $l1$ can dominate $l2$.

2.2 Comparisons

The most recent contributions using label setting algorithms to solve the TDSP forbid the night work. Under this assumption the algorithms speed up the solution process, since it is a more constrained version of problem; however, some instances could become infeasible, for example, when the time windows for a given customer are close or during the night intervals.

In order to test the performance and the quality of the solutions retrieved by the label setting algorithm, it is compared with a MILP model, without the night constraint (No-Night) and including it (Night). The total number of instances is 40 and the size of the instances varies between 3 and 15 customers. The entire set of instances and solutions are available at https://perso.isima.fr/~igpenaar/Odysseus_2021/. All the experiments have been achieved on an Intel® Core™i5-8400 at 2.81 GHz under Windows 10, using C++ and Gurobi 8.1.1. Table 1 presents the comparison between models.

	No-Night			*Night*		
Model	**Avg. Running time (s)**	**Speed factor**	**Infeasible**	**Avg. Running time (s)**	**Speed factor**	**Infeasible**
MILP	386.6	1183.1	0	737.4	97.5	7
LS	0.3	1	0	7.6	1	7

Table 1. Comparison between models.

Due to the night work constraint there is an increment in the running times, in the case of the MILP is about the double and for the label setting it rockets to 23 times. The label setting algorithm achieves the optimal solution provided by the MILP model for all instances in the No-Night case and over the 33 feasible instances in the Night case. The average running time is 737.4 seconds for the MILP, and 7.5 seconds for the label setting algorithm. Moreover, the label setting algorithm is 97 times faster than the MILP model.

3 Conclusions

Considering the complete set of rules from the EC social legislation increases the computational effort for both methods, notably for the MILP model on some instances. The label setting algorithm has a good performance in comparison with the MILP version, even though better computational times should be required to use this algorithm in an integrated vehicle routing procedure.

References

[1] A. Goel and T. Vidal, "Hours of service regulations in road freight transport: An optimization-based international assessment", *Transportation Science* 48(3), 391-412, 2014.

[2] A. Goel and S. Irnich, "An exact method for vehicle routing and truck driver scheduling problems", *Transportation Science* 51(2), 737-754, 2016.

[3] A.L. Kok, C.M. Meyer, H. Kopfer, J.M. Schutten, "A dynamic programming heuristic for the vehicle routing problem with time windows and European Community social legislation", *Transportation Science* 44(4), 429-553, 2010,

[4] M. Drexl, E. Prescott-Gagnon, "Labelling algorithms for the elementary shortest path problem with resource constraints considering EU drivers' rules", *Logistics Research* 2(2), 79-96, 2010.

[5] A. Goel, "Legal aspects in road transport optimization in Europe", *Transportation Research part E: Logistics and Transport Review* 114, 144-162, 2018.

[6] C. Tilk and A. Goel, "Bidirectional labeling for solving vehicle routing and truck driver scheduling problems", *European Journal of Operational Research* 283(1), 108-124, 2020.