



**HAL**  
open science

# A mixed integer programming formulation for the truck drivers scheduling problem considering the European Union drivers rules

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

## ► To cite this version:

Iván Peña-Arenas, Thierry Garaix, Philippe Lacomme, Nikolay Tchernev. A mixed integer programming formulation for the truck drivers scheduling problem considering the European Union drivers rules. 2021 IEEE 17th International Conference on Automation Science and Engineering, CASE 2021., Aug 2021, Lyon, France. IEEE, pp.101-106, 2021, 978-1-6654-1873-7. 10.1109/CASE49439.2021.9551634 . hal-03726184

**HAL Id: hal-03726184**

**<https://uca.hal.science/hal-03726184v1>**

Submitted on 18 Jul 2022

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Copyright

# A mixed integer programming formulation for the truck drivers scheduling problem considering the European Union drivers rules

Iván Peña-Arenas, Thierry Garaix, Philippe Lacomme, and Nikolay Tchernev

**Abstract**—The legal driver rules in the European Union defines a general framework with restrictions in driving time or working time between breaks and rest periods. These constraints must be addressed for an evaluation of any vehicle trip that met the regulation rules. For many real-world applications, the final trips have to satisfy these rules. The set of EU rules is the most complex design of trips and extension to other rules should be easy following the model we propose here. Compared to previous contributions we provide a new mixed integer linear program, which includes all the weekly rules to schedule when the sequence of visits is fixed. We also provide a new benchmark with detailed optimal solutions. Based on a set of numerical experiments, we discuss the relevance of different simplifications in the model used in the literature.

## I. INTRODUCTION

In the European Union, road freight transport must comply with several legal requirements including speed limits and access restrictions for example; the optimization framework could precompute and easily include these requirements because they focus only on the distance matrix [1]. Considering the European drivers rules in the Truck Drivers Scheduling is more complex. Different approaches have been used to include respite periods within the vehicle scheduling, considering different regulation rules [2]-[5]. Several linear formulations for the Truck Driver Scheduling Problem under different regulations have been proposed [6]-[9]. Linear formulations of such problems are useful, to evaluate sequences of visits but also as a part of a solution approach designed for routing problems [10]-[11].

Short respite periods are named breaks and long periods rests (more than 9 hours). A shift is the period starting at the first activity after a rest and finishing at the beginning of the next rest. The working time includes driving periods and service activities at the customer location or the depot (loading/unloading, cleaning the vehicle, etc). Other periods are called periods of availability (POA) or waiting time, meaning that the driver is neither working nor taking a break.

In this paper, the following set of constraints from the EU regulation are considered:

- R1. A driver cannot drive a total amount of 4.5h without a break of more than 45min referred to as "full break".
- R2. A full break of 45min can be split into two periods, a first break of at least 15min (and less than 45 min) and a second break of at least of 30min; the breaks should be taken in this order.

- R3. If the total working time of a shift is less than 6h, a break of at least 15 minutes has to be scheduled.
- R4. If the total working time of a shift is between 6h and 9h, a break of 30 min has to be scheduled.
- R5. If the total working time during a shift is greater than 9h, a break of 45 min is required.
- R6. The maximal driving time in a shift is 9h that could be extended to 10h twice per week.
- R7. A shift cannot exceed 24 hours.
- R8. The minimal rest period is 11h, that could be reduced to 9h three times per week. Therefore, the sum of breaks, POA and working durations is bounded by 13h; 15h if the rest is reduced to 9h.
- R9. A rest can be taken into two periods, the first an uninterrupted period of at least 3h, and the second a rest of at least 9h.
- R10. If the shift includes night working time, then the total working time in the shift is limited to 10h.

To the best of our knowledge, different rules have not been previously included in linear formulations, such as: split rests, working breaks and night work [6]-[9]. In addition, a previous work have proposed a different assumption on the night work rule, but the night rule of the EU regulation [1]. The paper is organized as follows. A literature review on the Truck Driver Scheduling Problem (TDSP) is provided in Section II. Section III presents the assumptions and the linear formulation. A series of experiments on the MILP are developed in Section III. Finally, in section IV conclusions are given.

## II. LITERATURE REVIEW

Different approaches have been used to include breaks and rest periods within the vehicle scheduling, considering different regulation rules. Some of them are interested in verify the feasibility of a given schedule. Archetti and Savelsbergh [2] proposed an algorithm with a complexity of  $O(n^3)$  to check feasibility of a driver schedule. The sequence of pick-up and delivery request is done considering time windows at the pick-up locations and the United States Hours of Service. Later, Goel and Kok [3] under the same regulation rules, developed a method which guarantees to find feasible driver schedules if they exists. This method solves the problem in  $O(n^2)$  in the case of single time windows. In the event of multiple time windows, the same complexity could be attained if the gap between subsequent time windows of the same location is at least 10 hours.

Goel [4] presents a breadth-first search (BFS) algorithm to find feasible truck driver schedules if they exist under the working hours regulation for the EU. The experiments use sequences of customers varying from 3 to 12, and the results show that previous methods fail to find feasible solutions half of the time when the size of the instances is large. Another result, is that the possibility of taking breaks in two parts does not bring a big difference for long distance hauling. Considering the Australian standard rules and basic fatigue management, Goel et al [5] presented an exact algorithm to solve the truck driver-scheduling problem. In addition, they presented four heuristics, based on removing some of the most computationally expensive steps of the exact method. As a result, the computational time diminishes, albeit, they did not guaranteed feasibility in all cases.

Among linear model formulations of the TDSP, Goel [6] developed a linear model complying with the Australian Heavy Vehicle Fatigue Law, in which the objective function is to minimize the total duration of the schedule. Drivers may only take rest periods before the service at a given location, albeit, they include parking lots or dummy locations with zero working time, to take rest periods after completing the service.

Taking under consideration the Canadian Hours of Service, Goel et al [7] find the set of feasible schedules for the Canadian Truck Driver Scheduling Problem (CAN-TDSP) using two heuristics and one enumerative method, the three of them based on a dynamic programming approach. The two heuristic approaches presents a good performance in terms of computational time, while, the enumerative method is not competitive under these criteria. Later, Goel [8] presents a mixed integer programming formulation for the Canadian minimum duration truck driver-scheduling problem (CAN-MDTDSP). The model is solved using a commercial solver CPLEX 12 and compare its results, against an iterative version of the dynamic programming algorithm presented in Goel et al [7]. Even though, CPLEX 12 offers better solutions while solving the mixed integer program, the iterative dynamic programming approach requires significantly smaller computational time.

Previous linear formulations considering the EU regulation have been proposed. The minimum duration truck driver-scheduling problem (MD-TDSP) is a generic model adapted to consider both, the EU and the US Hours of Service regulations proposed by Goel [9]. Kok et al [10] present the Vehicle Departure Optimization (VDO) with time-dependent travel times, which is a post-processing step of the VRPTW used as a means to verify the feasibility of a route. Both formulations, assume that breaks and rest periods are taken only at customers or suitable locations, which is not the general case.

Most recent dynamic programming models to generate compliant schedules under EU regulation are proposed by [1] and [12]. Even though, they consider more rules within their models, they propose a simplified version of the night working rule, assuming that a rest is taken every night, and do not offer optimal solutions for the TDSP.

To sum up, all MILP formulations work under the assumption of taking breaks at customers or suitable

locations, meaning that they work under non preemptive assumptions, and do not consider different rules from the EU regulation, particularly the night rule. On the other hand, other algorithms provide feasible schedules considering simplifications of the night working rule or even not considering it at all.

### III. MIXED INTEGER LINEAR PROBLEM

#### A. Data and variables

The initial set of data gives the sequence of locations to visit with the traveling times and the working times at each location. Time window constraints are imposed on the starting time of services at customer locations.

Each activity is modeled through a set of nodes. At each node a processing time and a respite period can be assigned in this order. The number of nodes is bounded according to the type and the duration of the activity, see table I.

Figure 1 gives an example of a sequence to evaluate, which has 3 customers and includes the depot at both, first and last position. The set of activities  $A$  corresponds to the service and transport activities developed at each location. As well, a fixed number of nodes assigned to each activity composes the set of nodes  $U$ .

In order to retrieve the activity  $a$  related with a node  $u$ , also the first and the last node  $u$  related with an activity  $a$ , three vectors are defined  $a(u)$ ,  $fu(a)$  and  $lu(a)$ , respectively.

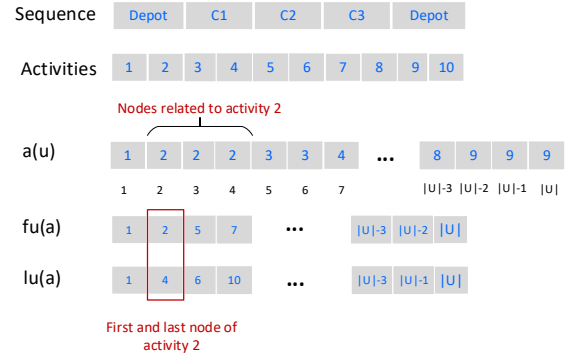


Figure 1. Modeling data structure.

All the parameters of the model are listed and the linear constraints are given just after. Note that all durations are in hours.

#### Sets

$A$ : Driving and service activities.

$U$ : Nodes to schedule breaks.

$B = \{15min, 30min, 45min, 3h, 9h, 11h\}$ : Types of break.

$I = \{[0,4], [24,28], \dots, [144,148]\}$ : Night intervals within a week.

#### Parameters:

$M$ : A big number.

$fu(a)$ : First node related to activity  $a$ .

$lu(a)$ : Last node of activity  $a$ .

$cp_a$ : Duration of activity  $a$ .

$\delta_u$ : 1, if node  $u$  is driving; 0, otherwise.

$a(u)$ : Activity related with the node  $u$ .  
 $e_a$ : Earliest starting time of the activity  $a$ .  
 $l_a$ : Latest starting time of the activity  $a$ .  
 $BD_b = \{0.25, 0.50, 0.75, 3, 9, 11\}$ : Minimal break durations.  
 $LB_i = \{0, 24, \dots, 144\}$ : Upper bound of the night interval  $i$ .  
 $UB_i = \{4, 28, \dots, 148\}$ : Upper bound of the night interval  $i$ .  
 $WL9$ : Break duration if working time is less than 9h [30min].  
 $WG9$ : Break duration if working time is greater than 9h [45min].  
 $CWT$ : Continuous working time without a break [6h].  
 $WSB1$ : Maximal working time in a shift without at least  $WL9$  (= 30) minutes of break [6h].  
 $WSB2$ : Maximal working time in a shift without at least  $WG9$  (= 45) minutes of break [9h].  
 $CDT$ : Continuous driving time without a break [4.5h].  
 $MDR$ : Regular driving time per shift [9h].  
 $MDE$ : Extended driving time per shift [10h].  
 $DE$ : Maximum number of driving extensions per week [2 times].  
 $RE$ : Maximum number of rest reductions per week [3 times].  
 $DDU$ : The maximal duration without a rest [24h].  
 $MWN$ : Maximum working time if night work is performed [10h].

Decision variables:

$x_u$ : Starting time of process at node  $u$ .  
 $y_u^b$ : 1, if at node  $u$  a break of type  $b$  takes place after the process. 0, otherwise.  
 $F_u$ : 1, if at node  $u$  a full break takes place. 0, otherwise.  
 $E_u$ : 1, if the driving time is extended during one hour. 0, otherwise.  
 $H_u$ : 1, if a rest of 9h is taken without a previous rest of 3h. 0, otherwise.  
 $b_{uv}$ : 1, if working time between nodes  $u$  and  $v$  is greater than 6h. 0, otherwise.  
 $c_{uv}$ : 1, if working time between nodes  $u$  and  $v$  is greater than 9h. 0, otherwise.  
 $t_{uv}$ : 1, if working time is performed during the night. 0, otherwise.  
 $r_{uv}$ : 1, if the starting time between two nodes  $u$  and  $v$  is less than  $DDU$  hours. 0, otherwise.  
 $CP_{ui}$ : 1, if the activity at node  $u$  starts before the upper bound of the interval  $i$ . 0, otherwise.  
 $CPP_{ui}$ : 1, if the activity at node  $u$  starts after the lower bound of the interval  $i$ . 0, otherwise.  
 $NPM_u$ : 1, if there is night work at node  $u$ . 0, otherwise.  
 $ti_{ui}$ : 1, if the activity at node  $u$  is developed at the night interval  $i$ . 0, otherwise.

Comfort variables:

$p_u$ : Processing time at node  $u$ .  
 $wo_{uv}$ : Working time between nodes  $u$  and  $v$ .  
 $d_{uv}$ : Driving time between nodes  $u$  and  $v$ .

## B. Objective function and constraints.

The objective function (1) is to minimize the completion time of the processing time of the last node.

$$\text{Min } z = x_{|U|} + p_{|U|} \quad (1)$$

Constraints (2) and (3) enforce the sequence between activities and nodes, respectively.

$$x_{fu(a)} + cp_a + \sum_{u=fu(a)}^{lu(a)} \sum_{b=1}^{|B|} BD_b * y_u^b \leq x_{fu(a+1)} \quad a = 1, \dots, |A| - 1 \quad (2)$$

$$x_u + \sum_{b=1}^{|B|} BD_b * y_u^b + p_u \leq x_{u+1} \quad u = 1, \dots, |U| - 1 \quad (3)$$

Only one type of break could take place at each node is enforced by constraints (4).

$$\sum_{b=1}^{|B|} y_u^b \leq 1 \quad a = 1, \dots, |A| \quad (4)$$

Constraints (5) ensure that service starts in the given time window.

$$e_a \leq x_{fu(a)} \leq l_a \quad a = 1, \dots, |A|; \delta_{n(a)} = 0 \quad (5)$$

Constraints (6) guarantee a full break after  $CDT$  hours of continuous driving time. Figure 2 depicts this situation.

$$d_{uv} \leq CDT + M \sum_{w=u}^{v-1} F_w \quad u, v \in U, u \leq v \quad (6)$$

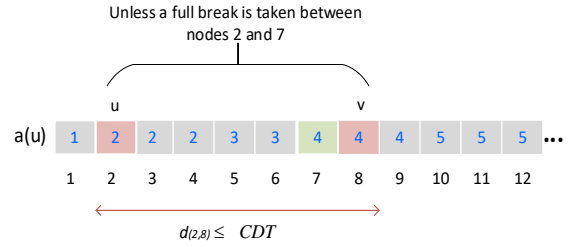


Figure 2. Compulsory full break after  $CDT$  hours of driving.

Constraints (7)-(9) enforce full breaks, when the break is greater or equal 45min ( $b \geq 3$ ), and forbid full break if the break is less than 30 min ( $b < 2$ ).

$$\sum_{b=3}^{|B|} y_u^b \leq F_u \quad u = 1, \dots, |U| \quad (7)$$

$$\sum_{b=2}^{|B|} y_u^b \geq F_u \quad u = 1, \dots, |U| \quad (8)$$

$$1 - y_u^1 \geq F_u \quad u = 1, \dots, |U| \quad (9)$$

There is not full break at  $v$  if a break of 15min or 30min is scheduled at  $v$ , and if there is not break less than 45 min between  $[u, v[$  (as the first part of the split) and a full break is scheduled just before  $u$ . This inequalities (10) forbids breaks of 30 min to be considered as full break if there is not first part of split break. Note that  $F_0 = 1$  by definition.

$$\left(1 - \sum_{b=1}^2 y_v^b\right) + \sum_{w=u}^{v-1} \sum_{b=1}^2 y_w^b + (1 - F_{u-1}) \geq F_v$$

$$v = 1, \dots, |U|; u = 1, \dots, v; \quad (10)$$

Working time between two nodes has to be less than six hours if there is no break in between them, as Constraints (11) apply.

$$wo_{uv} \leq CWT + M \sum_{w=u}^{v-1} \sum_{b=1}^{|B|} BD_b * y_w^b$$

$$u, v = 1, \dots, |U|, u \leq v \quad (11)$$

In Constraints (12) and (13) between two rests before  $u$  and after  $v$ , the sum of breaks has to exceed 30 (45 resp.) min. if the working time exceeds 6 h. ( $b_{uv} = 1$ ) (resp. 9h. and  $c_{uv} = 1$ ).

$$M \left( 2 - \sum_{b=5}^{|B|} y_v^b - y_{u-1}^b \right) + \sum_{w=u}^{v-1} \sum_{b=1}^{|B|} BD_b * y_w^b + M(1 - b_{uv})$$

$$\geq WL9$$

$$u, v = 1, \dots, |U|, u \leq v \quad (12)$$

$$M \left( 2 - \sum_{b=5}^{|B|} y_v^b - y_{u-1}^b \right) + \sum_{w=u}^{v-1} \sum_{b=1}^{|B|} BD_b * y_w^b + M(1 - c_{uv})$$

$$\geq WG9$$

$$u, v = 1, \dots, |U|, u \leq v \quad (13)$$

Binary indicators  $b_{uv}$  and  $c_{uv}$ , which are set to one when the working time exceed the their respective limits.

$$M * b_{uv} \geq wo_{uv} - WSB1 \quad u = 1, \dots, |U|; u \leq v \quad (14)$$

$$M * c_{uv} \geq wo_{uv} - WSB2 \quad u = 1, \dots, |U|; u \leq v \quad (15)$$

Constraints (16) limit the driving time between  $u$  and  $v$  to  $MDR$  plus the possible extension if a rest precedes  $u$  and there is not rest until  $v$ . Constraint (17) limit the number of extensions. In addition, note that  $y_0^b = 1$  by definition.

$$d_{uv} \leq MDR + E_u + M \left( 1 - \sum_{b=5}^{|B|} y_{u-1}^b + \sum_{w=u}^{v-1} \sum_{b=5}^{|B|} y_w^b \right)$$

$$u, v = 1, \dots, |U|, u \leq v \quad (16)$$

$$\sum_{u=1}^{|N|} E_u \leq DE \quad (17)$$

As Figure 3 shows, Constraints (18) enforces that between the finishing time plus the break of node  $v$  and the starting time of node  $u$  there is less than  $DDU$  hours, if there is not a rest between  $u$  and  $v - 1$ .

$$x_v + p_v + \sum_{b=1}^{|B|} BD_b * y_v^b - x_u \leq DDU + M \sum_{w=u}^{v-1} \sum_{b=5}^{|B|} y_w^b$$

$$u, v = 1, \dots, |U|, u < v \quad (18)$$

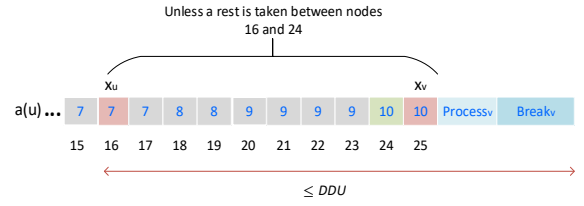


Figure 3. Compulsory rest each  $DDU$  hours.

Constraints (19) and (20) compute the processing time at node  $u$ .

$$p_u = x_{u+1} - x_u + \sum_{b=1}^{|B|} BD_b * y_u^b$$

$$a = 1, \dots, |A|; u = fu(a), \dots, lu(a) - 1 \quad (19)$$

$$p_{lu(a)} = cp_a - \sum_{u=fu(a)}^{lu(a)-1} p_u$$

$$a = 1, \dots, |A| \quad (20)$$

Constraints (21) and (22) compute the driving time and the working time between two nodes  $u$  and  $v$ , respectively.

$$d_{uv} = \sum_{w=u|\delta_u=1}^v p_w \quad u, v = 1, \dots, |U|, u < v \quad (21)$$

$$wo_{uv} = \sum_{w=u}^v p_w \quad u, v = 1, \dots, |U|, u < v \quad (22)$$

Constraints (23) establish the conditions for a split daily rest. There is a rest of 9h at node  $v$  and a rest at node  $u$ , if between  $u, v$  a break of 3h takes place, the rest of 9h taken at  $v$  is not a reduced rest, but the first part of a split daily rest.

$$y_v^5 + \sum_{b=5}^{|B|} y_u^b - \sum_{w=u+1}^{v-1} \sum_{b=4}^{|B|} y_w^b \leq H_v + 1$$

$$u = 1, \dots, |U|; u \leq v \quad (23)$$

Constraints (24) limits the number of reduced rests ( $y_u^5 = 1$ ) up to  $RE$ .

$$\sum_{u=1}^{|U|} H_u \leq RE \quad (24)$$

*Goel night rule constraints (25)-(27): night working is forbidden.*

Constraints (25) and (26) indicate if a node  $u$  starts before the upper bound  $UB_i$  or after the lower bound  $LB_i$  of interval  $i$ , respectively.

$$M * CP_{ui} \geq UB_i - x_u \quad u = 1, \dots, |U|; i = 1, \dots, |I| \quad (25)$$

$$M * CPP_{ui} \geq x_u + p_u - LB_i \quad u = 1, \dots, |U|; i = 1, \dots, |I| \quad (26)$$

Constraints (27) forbid night work.

$$CP_{ui} + CPP_{ui} = 1 \quad u = 1, \dots, |U|; i = 1, \dots, |I| \quad (27)$$

EU night rule constraints (25)–(26) and (30)–(32): Night working limits shift working time to 10h.

Constraints (28) enforce the working time between  $u$  and  $v$  to be less than  $MWN$  hours, if between the starting time of  $u$  and the finishing time of  $v$  some night work ( $t_{uv} = 1$ ) has been performed and there are less than  $DDU$  hours elapsed between  $u$  and  $v$  ( $r_{uv} = 1$ ). The value of  $r_{uv}$  is set by Constraints (32). The value of  $t_{uv}$  is set by Constraints (29)–(31) where  $t_{ui}$  means that  $u$  starts before the upper bound  $UB_i$ , finishes after the lower bound  $LB_i$  and the processing time is greater than 0.

$$wo_{uv} \leq MWN + M[2 - t_{uv} - r_{uv}]$$

$$u, v = 1, \dots, |U|; u \leq v \quad (28)$$

$$M * t_{uv} \geq \sum_{w=u}^v t_{ui} \quad u, v = 1, \dots, |U|; \quad (29)$$

$$u \leq v; i = 1, \dots, |I|$$

$$t_{ui} \geq CP_{ui} + CPP_{ui} + NPM_u - 2 \quad u = 1, \dots, |U|; \quad (30)$$

$$i = 1, \dots, |I|$$

$$M * NPM_u \geq p_u \quad u = 1, \dots, |U|; \quad (31)$$

$$i = 1, \dots, |I|$$

$$M * r_{uv} \geq DDU - [x_v - x_u] \quad u, v = 1, \dots, |U|; \quad (32)$$

$$u \leq v$$

### C. Assumptions and improvements.

Following the EU rules, we made some assumptions. Each activity can be split into an unlimited number of pieces of any duration. Several breaks or rests can be processed without any working time in between. Break and rest periods durations are not restricted to a limited set of values. Without loss of generality, the schedule starts and ends with a rest at the depot.

We compute an upper bound on the number of nodes per activity depending on its type (service or transport) and duration. If a driving activity is less than 4.5h then optimality requires 1 split of the activity for a rest or no split and 1 additional split for the night, in some cases to cover the night it is necessary to schedule more than one rest or break consecutively. It is easy to state that any break can be scheduled before or after the activity without loss of generality. Similar reasoning gives Table 1 for different values of service and driving activities. The number of nodes is the number of nodes for split plus one. For long activities, some split nodes model rest and break for a shift included in the activity.

We identified the longest possible sequence of breaks in an optimal solution: a break ( $< 9h$ )  $\rightarrow$  a rest ( $\geq 9h$ )  $\rightarrow$  the first part of split break ( $3h$ )  $\rightarrow$  the first part of split break ( $< 45min$ ). The first break is imposed by respite at of the first shift. The two last breaks may initiate split rests and breaks. We can show that such series of breaks and rest do not have to be scheduled in optimal solution only at the beginning of an activity. Therefore, the number of nodes per activity is

increased by 3 and the processing times of the three first nodes of an activity are set to zero.

TABLE I. MAXIMUM NUMBER OF NODES BY TYPE OF ACTIVITY AND DURATION

Activity	Duration	Maximum number of nodes
Service	$< 6h$	3
	$< 13h$	4
	Otherwise	$\frac{cp_a}{3} + 3$
Transport	$< 4.5h$	3
	$< 9h$	4
	Otherwise	$\frac{cp_a}{3} + 3$

For R5 the cumulated duration of break can be reached by two breaks of 20 and 25 minutes. Therefore the duration of break could be optimally set with values different than lower bounds of break durations. An additional continuous variable  $D_u$  is ranged by break duration interval of the type of break scheduled at node  $u$ . However, minimal respite durations to satisfy constraints usually provide optimal solutions; one example (Instance 41) has been built where optimality is not reach in this case.

Finally, the number of constraints is reduced by limiting the pairs of nodes  $u$  and  $v$  in the domain of the constraints for which the working or driving time between their activities ensures that the constraint does not apply. For instance, constraint (6) should exclude the nodes for which the cumulated driving time between  $a(u) + 1$  and  $a(v) - 1$  is greater than 4.5h.

## IV. NUMERICAL EXPERIMENTS

### A. Instances

We propose a new set of instances PGLT available at [https://perso.isima.fr/~igpenaar/Lyon\\_IEEE\\_2021/](https://perso.isima.fr/~igpenaar/Lyon_IEEE_2021/). The total number of instances is 41 and the size of the instances varies from 3 to 15 customers. In addition, we derived one instance from each route of the solutions of the routing problem provided by [1]. This set of 157 instances is named GOEL.

### B. Results

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. A limit on the computation time has been imposed after two hours.

The first experiment aims to compare *EU-Setting* (model (1)–(26); (28)–(32)) against *G-Setting* (model (1)–(27)) over the two sets of instances. Note that the solutions obtained with *G-Setting* on GOEL instances differ from the solutions provided by a heuristic - according to our objective - in [1]. The improvement is more than one hour for 15.3% of the routes for one day to one week.

The results are presented in Table II and Table III. Table II presents four types of statistics for each of the two set of instances under the two set of settings. And in table III, *EU-*

*Settings* is the base scenario and its performance is compared against *G-Settings* by computing three different statistics. From Table II, solutions with both settings are close in average but Table III, shows that *EU-Setting* allows to find better schedules for one half of instances and the improvements can reach 8.5 hours. Moreover, for PGLT instances, *G-Setting* is infeasible for 7 instances which are feasible with *EU-Setting*. Nevertheless, *G-Setting* is solved faster (almost 100 times) than *EU-Setting* with our formulations.

TABLE II. EU-SETTING VS. G-SETTING

Statistic	PGLT		GOEL	
	EU-Setting	G-Setting	EU-Setting	G-Setting
Avg. completion time (h)	33.7	34.4	103.8	104.2
Avg. CPU time (s)	238.7	2.4	119.7	3.5
Max. CPU time(s)	7200.0	18.6	7200.0	35.3
Infeasible	7	14	0	0

TABLE III. MILP UNDER EU-SETTING PERFORMANCE

Statistic	PGLT	GOEL
Instances improved (%)	46.3	50.3
Avg. GAP (%)	1.7	1.4
Maximum difference(h)	3.2	8.5

The second experiment has two treatments. The First execute the MILP under EU-Setting using the completion time (COM) as objective function and compare its results adding the optimal makespan (UBM) as an upper bound constraint (COM+UBM). The second treatment uses the makespan (MKS) as objective function and compare its results adding the optimal completion time (UBC) as an upper bound constraint (MKS+UBC). This experiment only use the PGLT instances, because computational running times using GOEL were excessively high. Table IV presents the results.

TABLE IV. MILP UNDER EU-SETTING ADDING THE MAKESPAN AND THE COMPLETION TIME AS UPPER BOUNDS

Statistic	COM+UBM	MKS+UBC
Number of instances with different objective function value	9	9
Avg. GAP (%)	1.7	2.7
Max. GAP (%)	16.9	30.0

For both treatments, in nine instances it was not possible to achieve the same optimal value obtained without the upper bound constraint. The quality of the solution in both cases diminishes 1.7% and 2.7% on average and in the worst case, this reduction could be of 16.9% and 30.0% respectively. In this regard, when minimizing the completion time it is not

possible to guarantee a solution with the minimum makespan and the opposite.

## V. CONCLUSION

A new MILP formulation for the truck driver scheduling problem under the European Union drivers rules is presented. The model extends previous formulations considering rules like: split daily rests, working breaks and night work. Two different assumptions for the night work rule are considered, in order to compare its results with a previous algorithm in the literature. Results show a good performance obtaining equal solutions in 66.2% of the instances. Different experiments indicates that there is not a significant difference in the solutions between the two assumptions for the night constraint, even though, Goel night assumption is more flexible with the drivers. In addition, the Goel night rule assumption highly improves the total execution time of the model. Finally, the model results could lead to develop handcrafted algorithms in order to solve efficiently the problem.

## REFERENCES

- [1] Goel, A., "Legal aspects in road transport optimization in Europe", *Transport. Res. E-Log*, vol.114, pp. 144–162, 2018.
- [2] Archetti, C. and Savelsbergh, M.W.P., "The trip scheduling problem", *Transp. Sci.*, vol. 43, pp. 417–431, 2009..
- [3] Goel, A. and Kok, A.L., "Truck driver scheduling in the United States", *Transp. Sci.*, vol. 46, pp. 317–326, 2012.
- [4] Goel, A., "Truck driver scheduling in the European Union", *Transp. Sci.*, vol. 44, pp. 429–441, 2010
- [5] Goel, A., Archetti, C., and Savelsbergh, M.W.P., "Truck driver scheduling in Australia", *Comput. Oper. Res.*, vol. 39, pp. 1122–1132, 2012.
- [6] Goel, A., "A mixed integer programming formulation and effective cuts for minimising schedule durations of Australian truck drivers", *J. Sched.*, vol. 15, no. 6, pp. 733–741, 2012..
- [7] Goel, A. and Rousseau L-M., "Truck driver scheduling in Canada", *J. Sched.*, vol.15, pp. 783–799, 2012.
- [8] Goel, A., "The Canadian minimum duration truck driver scheduling problem". *Comput. Oper. Res.*, vol. 39, no. 10, pp. 2359–2367, 2012.
- [9] Goel, A., "The minimum duration truck driver scheduling problem", *EURO J. Transp. Logist.*, vol. 1, no. 4, pp. 285–306, 2012.
- [10] Kok, A.L., Hans, E.W., and Schutten, J.M.J., "Optimizing departure times in vehicle routes", *Eur. J. Oper. Res.*, vol. 210, pp. 579–587, 2011.
- [11] Koç, C., O. Jabali, and G. Laporte. "Long-haul vehicle routing and scheduling with idling options", *J. Oper. Res. Soc.*, vol.69, no. 2, pp. 235–246, 2018.
- [12] Tilk, C. and Goel, A. "Bidirectional labeling for solving vehicle routing and truck driver scheduling problems", *Eur. J. Oper. Res.*, vol. 283, pp. 108–124, 2020.