TRANSPORT
ISSN 1648-4142 / eISSN 1648-3480
2021 Volume 36 Issue 1: 84–97
https://doi.org/10.3846/transport.2020.13784

DEFINITION OF A PYTHON SCRIPT FOR THE MICRO-SIMULATION OF THE TRUCK PLATOONING SYSTEM

Serio AGRIESTI[1*], Marco PONTI[2], Luca STUDER[3], Roberto MAJA[4],
Ornella ALECCIA[5], Jochen LOHMILLER[6], Isaac Vargas GORDILLO[7]

[1]Dept of Built Environment, Aalto University, Espoo, Finland
[2, 3, 4, 5]Mobility and Transport Laboratory, Dept of Design, Polytechnic University of Milan, Milan, Italy
[6, 7]PTV Group, Karlsruhe, Germany

Submitted 13 July 2019; resubmitted 6 March 2020, 18 March 2020; accepted 20 May 2020;

Abstract. Truck platooning is, by now, one of the major topics in transport science and freight transport. The benefits arising from the system explain the growing interest of the involved stakeholders and the many field-tests planned in the next years. This run towards truck platooning saw an abrupt acceleration but there are risks that should be accounted for. Even though field-tests are fundamental for the implementation of a new transport system, they will hardly cover all the traffic scenarios that a platoon of trucks will face on the European network. Therefore, there is the need for many more studies based on traffic simulation and for tools enabling traffic simulation software to reproduce truck platooning. In this framework, the paper has two aims, the first one being to report and describe a *Python* script to reproduce truck platooning with a common commercial simulation software. The second one is to apply said script to analyse what is the best driving strategy for a platoon of truck to limit the hindrance on the surrounding traffic while approaching a critical highway segment such as the on-ramp one. At the end of the paper, a comparison between three different strategies (driving as usual, dissolution and headway adaptation) is carried out and commented.

Keywords: truck platooning, *Python* script, impact assessment, C-Roads, traffic efficiency, traffic modelling.

Introduction

The truck platooning system exploits the Cooperative Adaptive Cruise Control (CACC) in order to compose a platoon of heavy vehicles, travelling almost with the same speed and the same driving regime. In the short time horizon (2019–2025) the human driver will still be entrusted with the lateral driving task, while the longitudinal control will be granted to the on-board L1 system (Ricardo UK Ltd 2014; Bishop 2017; EC 2018). This solution should allow the heavy vehicles to travel with a strongly reduced time gap between them, keeping a value that can range between 0.3 and 1 s. (Studer *et al.* 2019), and to generate the following benefits: lower fuel consumption, increased road capacity, shockwave damping, increase of traffic flow efficiency, etc. Moreover, the increasing connectivity of freight vehicles is a process already begun and with different proved applications, everyday fleet management is in fact carried out through connectivity and Intelligent Transportation System (ITS) solutions such as, for example, the ones described by Benza *et al.* (2012).

When considering all of the above, it is clear why many of the involved stakeholders consider truck platooning as inevitable, especially when it is acknowledged that this system is the first of many steps towards complete automation (ERTRAC 2017). The potential benefits are relevant indeed, but the rush towards the implementation of the system carries some risks. Aim of this paper is to provide a tool to prevent at least the risk of implementing the system without having examined in deep the potential impacts (both positive and negative) of truck platooning on the overall traffic flow. In fact, as for the knowledge of the authors, currently there is not a traffic simulation software that integrates the truck platooning system as a default mode of transport. Therefore, the authors identified this lack of a common tool, easy to use and usable by everyone, as a gap to be filled in order to provide a shared algorithm and promote in literature simulation studies that are comparable and replicable. In fact, for every evaluation activity there should be an intermediate phase between

---

*Corresponding author. E-mail: *seriangelo@gmail.com*

the ex-ante and the ex-post ones in which a transport system is analysed through modelling works or other surrogate evaluation methodologies (Studer *et al.* 2019; Agriesti *et al.* 2018b). Moreover, the tool described is applied to assess what is the best driving strategy a platoon should follow to minimize its impact at an on-ramp area both as proof of concept and to obtain first valuable results aimed at enriching the current literature. The second objective is, in fact, to frame through Key Performance Indicators (KPIs) related to traffic efficiency, if the overall traffic flows benefits by a platoon keeping its intended headway (driving as usual), by a platoon that increases said headway to allow merging vehicles or by a platoon that dissolves itself upstream the ramp.

The paper is structured as follows: in Section 1 a short bibliographical review is carried out to identify the current capabilities of traffic simulation software and all the behaviours that should be included in a traffic simulation; in Section 2 an overview of the input parameters to be inserted through the *COM interface* and of how this interface is exploited is presented; moreover, in Section 3 the main scripted functions are reported and explained in their operational logics; in Section 4 the developed tool is exploited to evaluate three platooning driving strategies and their impacts on traffic efficiency, then in the last section the conclusions are presented.

1. Bibliographical review and current modelling approaches

In the short term, the main potentialities of the truck platooning system are bound to the strongly reduced headway between the vehicles that grants aerodynamic benefits and allows to decrease fuel consumption for all the following vehicles (Ricardo UK Ltd 2014; Brizzolara, Toth 2016; Lammert *et al.* 2014; Bakermans 2016). This benefit arises because a reduced space between heavy vehicles hinders the creation of vortexes and reduces the air drag. The range of possible reduction in fuel consumption can be quite wide – 5…20% (Liang *et al.* 2014) and changes with the position of the single vehicles within the platoon. This benefit is bound to the number of kilometres that the vehicles can drive in formation, thus a first KPI that should arise from simulations is how many times a platoon dissolves due to external factors; particularly relevant in this is the presented case study: it is clear how the dissolution strategy can impact on this KPI over a high number of kilometres.

Another benefit, enjoyed by the traffic flow as a whole, is the reduced space occupied by the platoon on the road due to the reduced headway granted by the CACC. In fact, as mentioned by Sia Partners (2016), reducing the time gap from 1.5 to 0.5 s can lead to a reduction of the spatial gap equal to 22 m and to a resulting increase by around 30% of road capacity. This benefit too is bound to the number of kilometres that the platoon drives in formation; it should be highlighted though how the benefits of truck platoons driving in formation must be compared with the potential negative impacts arising from the following manoeuvres: dissolving the platoon, responding to a cut-in[1], adapting the headway and reacting to changes in boundary conditions such as roadworks ahead (Deng 2016; Andersson *et al.* 2017; Agriesti *et al.* 2018a); two of which will be evaluated in Section 4. All these manoeuvres and possible interactions with the surrounding traffic should be formalized in research and sub-research questions (Studer *et al.* 2019; Agriesti *et al.* 2018a) and evaluated before the field-test phase or, at least, before the large-scale deployment of the system.

In traffic science, whenever automation is involved, one of the main arising criticalities is the interaction between the system and the surrounding traditional traffic. Truck platooning is no exception and how it interacts with the infrastructure and the traffic flow must be analysed in all the relevant scenarios. The replication of all the needed scenarios is hardly achievable only through field-tests; therefore, a traffic simulation software should be able to reproduce manoeuvres like cut-ins, dissolution of the platoon and headway adaptation. It is clear that traffic modelling is not a substitute for field-tests on public roads; rather it should be conceived as a tool to upscale the field-test results to all the traffic scenarios that could be relevant to a complete evaluation of the system (Studer *et al.* 2019; Agriesti *et al.* 2018b).

Therefore, the need of having a common tool to reproduce the case studies arises, as it does the need of having different research bodies across Europe following the same formalized modelling approach while assessing their national reality and how it fits the needs of this new transport system. This paper wants to provide this tool in a way that is easy to integrate with the *PTV Vissim* micro-simulation software (*https://www.ptvgroup.com/en/solutions/products/ptv-vissim*) and is able to reproduce the behaviour of a platoon of trucks in a precise but also flexible way. The script presented in this paper includes, in fact, all the longitudinal behavioural adaptation that could be performed in real life and that emerged from the bibliographical review carried out by Agriesti *et al.* (2018b).

In bibliography, some studies facing these issues can already be found. In research by Van Maarseveen (2017), for example, the impacts of truck platooning at on-ramp areas are analysed through modelling. A constant time gap regulation strategy was defined and two set of simulation were carried out, one considering the headway adaptation and one considering the lane change of the platoons of trucks. Another study that analyses through modelling the impacts of truck platooning near a ramp is research by Deng (2016). In this study, the author considered a platoon of 5 vehicles needing 150 s to dissolve reaching

---

[1] A cut-in maneuver is performed by a vehicle external to the platoon, trying to merge within the platoon from another lane. The vehicle causing a cut-in is not equipped to receive information passed through the CACC, so it cannot adapt its driving like the platooned vehicles do and therefore it causes the dissolution or the separation of the platoon itself.

a headway wider than 100 m, starting from a value of 10 m. Through modelling, the author compared the dissolving scenario with the one where the platoon does not dissolve and possibly hinders the other vehicles in taking the off-ramp. In research by Deng (2016), the impacts of the platoon formation manoeuvre are analysed too, being strongly influenced by the surrounding traffic and impacting traffic efficiency, environment and safety in return. Different formation strategies are considered, involving different speed values for the leading vehicle (70, 75 and 80 km/h) and different conditions of the traffic flow (light, medium and congested). Through modelling, the delay in the formation manoeuvre is calculated.

The presence of platoons itself can have an impact on the traffic flow, depending on the algorithm implemented in the CACC. In research by Gordon (2015), in fact, different CACC parameters were simulated to understand the impact deriving from a potentially increased road capacity and from the shockwave damping effect of string-stable platoons (Ploeg *et al.* 2011). The simulations concerned an American Highway (I-85) and the input variables were the headway, the CACC penetration rate in the heavy vehicles market and the traffic volume. A relevant result that emerges is that a Market Penetration Rate (MPR) higher than 20% is needed to achieve significant improvements on traffic flow; this result is a good example of the usefulness of traffic simulation in defining also business cases and economic estimations. Still on the I-85 and on the I-285, simulations were carried out to identify the impacts of truck platooning on said corridors (Smith 2016). In this work, the considered level of automation seems to range between L3 and L4 (SAE International 2016). It should be highlighted that in literature, in the short term, the foreseen level of automation is L1, meaning that only the longitudinal control is going to be entrusted to the automation. Moreover, the platoons of trucks will be most likely be confined on the slow lane and a dissolution is going to be needed before a necessary lane change. The script presented in this paper reflects this kind of needs and a similar logic is designed within the case study simulations. Similar consideration about the level of automation were made in research by Ramezani *et al.* (2018), within which a micro-simulation model is used to assess the impact of CACC equipped heavy trucks on the traffic flow on the I-710 corridor.

Many approaches were followed to reproduce the truck platooning system by the means of modelling software, ranging through micro-, meso- and macro-models. Still, not having a common, shared and easy to use script can hinder the comparison of the results between studies. Moreover, the software houses are still in the process of optimizing their own product to the new technologies such as autonomous driving or truck platooning, so the process to reproduce these systems can be not straightforward and can discourage evaluator across Europe from developing simulation studies on truck platooning. In fact, even if some studies are already in literature, their number is still limited and many of them have not been replicated in other realities or with other boundary conditions. It is important that, before the large-scale implementation of the system, a higher number of simulation studies is carried out to analyse transport problems such as the one presented in Section 4.

2. The *PTV Vissim* tool and the *COM interface*

Within the evaluation activities of C-Roads Italy (*https://www.c-roads.eu/pilots/core-members/italy/Partner/project/show/c-roads-italy.html*), modelling works are needed in order to upscale the impacts of the limited fleet of truck platoons that is going to drive on public roads. To carry out analyses similar to the one presented in this paper, the *PTV Vissim* software was identified as the main tool due to the *COM interface* that allows the evaluator to modify the longitudinal behaviour of the vehicles with a high degree of freedom. Therefore, before reporting the case study and its results, a short overview of the functions implemented in the presented script and of the *COM interface* is needed to ensure replicability.

The *COM interface* of *PTV Vissim* takes as input the *Python* (*https://www.python.org*) script and some User Defined Attributes (UDAs), in order to replicate the intended behaviour (Table 1). It should be highlighted that, even if the behaviour is scripted and thus not accessible directly through the *COM interface*, the presence of the UDAs gives to the end-users the capability to tune and adjust the kind of platoon that is to be simulated. For example, one of the simplest UDAs that the script takes as input through the interface is the maximum number of vehicles forming a platoon, letting the evaluator simulate a platoon of 3 rather than 10 vehicles. In the following, each UDA defined and integrated in the script is presented and elaborated on its theoretical worth, also providing the most acknowledged values found in literature (when available) and a suggested value representing the figure based on which the script was designed. The first ones are the result of an extensive research effort carried out within C-Roads Italy and resulted by Agriesti *et al.* (2018b), a technical report that analyses more than 130 bibliographical references. The second ones are the one exploited to design and evaluate the case study in Section 4, based on the literature values and chosen as default for the script.

Again, the above value refers to Agriesti *et al.* (2018b) and the suggested ones are applied in the presented use cases, but these values should be adapted to the designed scenario and to the necessities of the evaluators. It should be highlighted how the *COM interface* has a limit: it allows to regulate only the longitudinal behaviour of a platoon of vehicles. Therefore, the vehicles engaged in a platoon should be limited to the slow lane, in order to avoid erratic lane changes that would not reflect the actual behaviour. This does not mean that the script hinders the vehicles equipped for platooning from changing lane, it simply means that the platoon dissolves before these vehicles reacquire the vehicle type “Heavy Gross Vehicle”, which is equal to a traditional truck, and can freely change lane.

Table 1. UDAs

NoOfVehicles	Maximum number of vehicles that can compose a platoon. Literature value: 2…10.
DesSpeed	Desired speed of the vehicles composing the generated platoon. Suggested value: 80 km/h. Literature value: 80 km/h.
LinkNo *LaneNo*	Identification Numbers (IDs) of the links and lanes on which the platoon is generated.
safeDist	Distance at standstill; this value is used to generate the platoons. The vehicles are loaded on the network with a distance between each other equal to *safeDist*. Suggested value: 4.20 m.
DesignHdwy	Value of the headway that the platoon keeps in standard driving conditions. In *PTV Vissim* it is the distance between the front of the following vehicle and the front of the leading one*. It should be highlighted that this value should be higher than the one achieved through the time gap imposed by the driving behaviour. Should that not be the case, the vehicle would fluctuate trying to reach the designed headway but limited by the time gap of its driving behaviour. Suggested value: 20.21 m. Literature value: 20.21…40.21 m. *Note*: *It should be noted that the default length of a heavy vehicle, in *PTV Vissim*, is equal to 10.21 m. In order to get the spacing between the front of a following vehicle and the rear of the preceding one, this value must be subtracted from the headway value.
DissHdwy	Value beyond which a dissolving vehicle can consider the disengaging manoeuvre finished. It means that a vehicle that wants to decouple from the one ahead will wait to reach a headway equal or higher than *DissHdwy* before restoring the traditional driving regime. This distance reflects the headway usually kept by a human driver and takes into account also a traditional field of view. Suggested value: 40.21 m. Literature value: 40.21…55 m (time gaps of 1.5…2 s at 80 km/h).
EndHdwy	Value of the headway beyond which two vehicles are unconnected. If a following vehicle finds itself so distant from the preceding one, it stops trying to close the distance and restores a traditional driving regime. Suggested value: 120.21 m (to account for what is stated in the footnote number 2). Literature value: 110.21 m.
vehicleType(*L*)(*F*)(*D*)(*R*)	From the vehicle, type depends the driving behaviour. The script can change the vehicle type for each vehicle in the platoon in one of these four categories: *L*: Leading vehicle; *F*: Following vehicle; *D*: Vehicle during its separation from the platoon; *R*: Following vehicle adapting its headway, for example to respond to a ramp.
MaxTimeTransition	The script considers two vehicles unconnected based on their headway (see *EndHdwy*). It can happen, though, that during congestion two vehicles drive so slowly that they never reach the *EndHdwy* value. Therefore, *MaxTimeTransition* imposes a threshold beyond which the human driver is considered able to re-enter the driving loop. Suggested value: 5…10 s. Literature value: –.
LeadSlowing *FollowingSpeedUp*	The proposed script allows to tune the close up manoeuvre in a way that the end-user is able not only to trigger a speed up in the following vehicles but also a slowdown of the leading vehicle. This would allow to replicate study by Deng (2016). Suggested values: *LeadSlowing* 5 km/h and *FollowingSpeedUp* 15 km/h (within the script these are net values). Literature value: leading speed 70 km/h and following speed 85 km/h (absolute values)
DecisionDistCloseUp	In a platoon of more than two vehicles, it can happen that after having the headway increase (for example due to a ramp), the vehicles in the middle have to decide if accelerate to close the space ahead or to keep their speed constant until the following vehicles close up. The decision is tuned through this parameter. Suggested value: 20 m. Literature value: –.
RampMinimumLength	Minimum length beyond which a connector is considered a ramp. This value should be tuned according to the analysed network. However, usually the difference between the length of connectors and the on-/off- ramps should be enough to avoid false-positives within the script. Suggested value: 250 m. Literature value: –.
RampAheadDist	Distance upstream the ramp where said ramp is signalled. This parameter allows the end-user to decide where the platoons of trucks should adapt their behaviour in order to respond to a ramp (e.g. through headway adaptation). Suggested value: 500…5000 m. Literature value: –.
LinkOfReception	While simulating a C-ITS message, even though the telematic aspects are relevant indeed, is the change in behaviour that an evaluator wants to reproduce in order to assess impact areas as traffic efficiency or safety. By defining the "triggering" link, the script allows the user to ideally define the range of the message and, thus, the start of the change in behaviour (namely, the start of *LinkOfReception*). Suggested values for cooperative messages: 1000…5000 m. Suggested values for vertical signalling: 700 m. Literature value: –.

Due to the format of a research paper, the modified script is not reported as a whole in the following section, but only the main additions are listed, with the scripted lines being reported only for a function as example. Consequently, a base knowledge of the *Python* programming language is desirable in order to integrate the functions defined in this paper with the original script provided with the installation of *PTV Vissim*.

3. Truck platooning script

In this section, the functions added and integrated in the *Platooning.py* script are reported. In fact, the original script was not conceived to reproduce the platooning of heavy vehicles and did not include the following functions, the need for which arose from the consideration reported in the previous section. Therefore, this section means to make the user able to implement the script him or herself and to carry out simulations that are comparable in their results with the ones planned within the C-Roads Italy activities. It is clear that, even if the boundary conditions are the same, it is impossible to compare two studies taking in input different behaviours for the truck platooning system. In the Figure 1 the workflow followed by the script is reported.

3.1. Integration

Some of the functions defined in this section must be defined through the *COM interface* while others are periodically recalled by the function *UpdatePlatoons*. These are: *CheckForCutins*, *SyncSpeed*, *EndCloseUp* and *AdaptHdwy*. They were included within the *UpdatePlatoons* function for their periodic nature, in fact they are not triggered but are simply called after a time interval defined by the user through the *COM interface*, as in Figure 2.

Most of the functions added and presented in this paper should be integrated within the *UpdatePlatoons* function by the addition of the following lines:

```
»» CheckforCutins(platoon);
»» vehChangeDesSpeed = SyncSpeed(platoon, vehChangeDesSpeed);
»» vehChangeDesSpeed = EndCloseUp(platoon, vehChangeDesSpeed);
»» AdaptHdwy(platoon).
```

This way, each one of these functions should be called periodically with the same time interval as the *UpdatePlatoons* function. Only the functions that should not be triggered that often are left outside this function and are added as stand-alone, these are: *RampAhead*, *DissolvePlatoons1* and *DissolvePlatoons2*. These should be explicitly added through the *COM interface*. Within this interface (Figure 1), all the UDAs reported in Table 1 can be defined as input values based on of the parameters that the evaluator wants to study. Moreover, the period parameter defines how many seconds pass before each function is called again. For the *DissolvePlatoon1* function, for example, this value was set at 5 s (equal to 50-time steps). This means that, depending on the second at which a platoon reaches the *LinkOfReception*, it can start the dissolution instantaneously or with a delay of 5 s at most. This should reflect the possible delay in human reaction, reproducing a certain stochastic behaviour and response to the received signalling. This function will be exploited in the analysed case study to test one of the three strategies: *driving as usual*, *headway adaptation* and *dissolution*. To foster replicability, in the following all the added section will be reported as block schemas, to illustrate the reproduced behaviour, while only dissolve platoon will be reported as script lines to foster readability. Still, all the following functions may be shared by the corresponding author upon request.

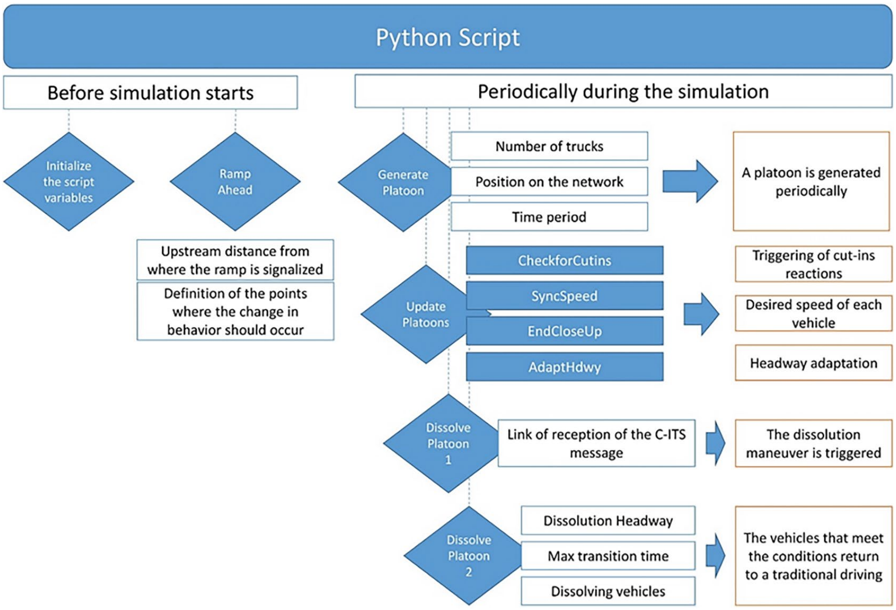
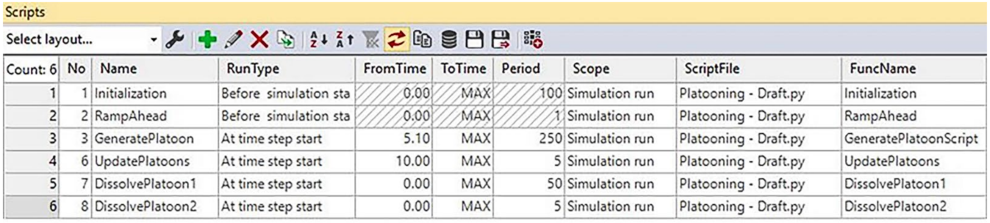


Figure 1. Script workflow



Count: 6	No	Name	RunType	FromTime	ToTime	Period	Scope	ScriptFile	FuncName
1	1	Initialization	Before simulation sta	0.00	MAX	100	Simulation run	Platooning - Draft.py	Initialization
2	2	RampAhead	Before simulation sta	0.00	MAX	1	Simulation run	Platooning - Draft.py	RampAhead
3	3	GeneratePlatoon	At time step start	5.10	MAX	250	Simulation run	Platooning - Draft.py	GeneratePlatoonScript
4	6	UpdatePlatoons	At time step start	10.00	MAX	5	Simulation run	Platooning - Draft.py	UpdatePlatoons
5	7	DissolvePlatoon1	At time step start	0.00	MAX	50	Simulation run	Platooning - Draft.py	DissolvePlatoon1
6	8	DissolvePlatoon2	At time step start	0.00	MAX	5	Simulation run	Platooning - Draft.py	DissolvePlatoon2

Figure 2. *COM interface*

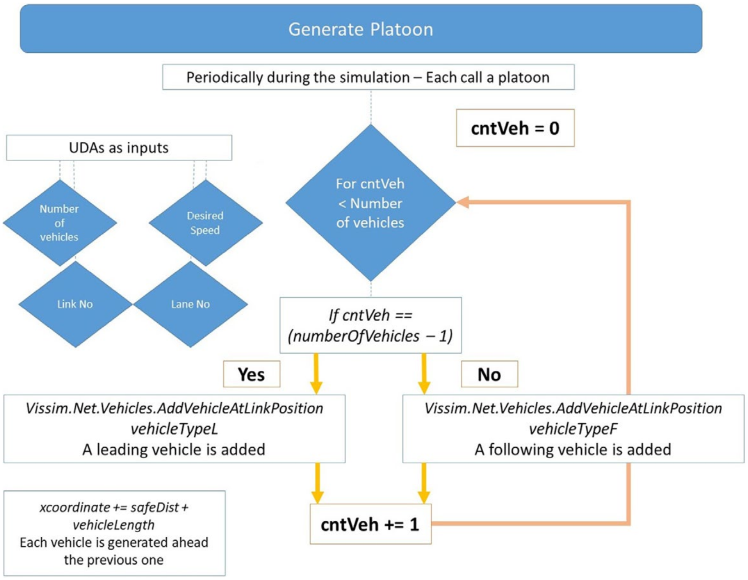


Figure 3. *Generate Platoon* function

3.2. Generate platoon

Generate Platoon function generates as much vehicles as intended, already platooned, outside of the traffic composition appearing from the different traffic inputs placed through the network (Figure 3).

3.3. *SyncSpeed*

SyncSpeed function is not in the original script and was added completely ex-novo. It synchronizes the platooned vehicles in a way that if the current headway is different from the *DesignHdwy* the vehicles adapt their speed to close or increase the gap. A similar function was already coded in the original script but did not allow to the end-user to define different speed values for the preceding vehicles and the ones behind.

This function was implemented to allow a more precise control of the speed for the different vehicles alongside the platoon, granting an increased freedom in simulating the closing up manoeuvre (Figure 4). In fact, the following vehicles can accelerate to reach the one ahead while it slows down to ease this manoeuvre. The actual speeds at which these manoeuvres are performed are decided through the UDAs.

3.4. *EndCloseUp*

The scope of *EndCloseUp* function is quite straightforward, once all the vehicles have restored their intended headway through *SyncSpeed*, the close up manoeuvre stops to condition desired speed value. This is accomplished through a simple for cycle checking the headway of the platooned vehicles every 0.5 s and verifying if they returned to the *DesignHdwy* UDA.

3.5. Dissolve platoon

Dissolve Platoon function is probably one of the core ones, when referred to the planned modelling activities of C-Roads Italy. In many of the defined use cases (Agriesti *et al.* 2018b), the reaction of a platoon to a cooperative message is its dissolution, triggered as soon as the message is received by the leading vehicle. Therefore, in order to simulate this behaviour, the following two functions were defined and added to the original script (by defining two different functions, it was made possible to define two different time period for their activation). These are reported to better illustrate the tool and how it is implemented both in the following case study and within the *COM interface* of *PTV Vissim* (see Appendix).

The function allows to define a different driving behaviour by changing the vehicle type. The first vehicle becomes instantaneously a traditional heavy vehicle, ready for example to perform a lane change as soon as it passes through the *LinkOfReception*. The following vehicles first become of *vehicleTypeD* type and try to recover a traditional headway value; this transition phase stops as soon as the intended headway is recovered or if a certain time threshold has passed. This way, all of the possible dissolution dynamics should be included in the transition phase, at the end of which the following vehicles become traditional ones, also able to change lane again.

3.6. *CheckforCutins*

CheckforCutins function is one of the most relevant in evaluating the interactions between platoons and the rest of the traffic, therefore it was added to the original script (Figure 5). Moreover, one of the three strategies tested in Section 4 is the headway adaptation that fosters cut-ins if needed by the merging vehicles; therefore, it was necessary to implement the corresponding behaviour within the script.

As mentioned above, different scenarios are accounted for:

- »» if the intruding vehicle cuts in just behind the leading vehicle, said vehicle becomes a traditional truck while the second one becomes the new leading vehicle;
- »» if the intruding vehicle cuts in somewhere between the second and the second-last truck, the platoon is split in two parts;
- »» if the intruding vehicle cuts in ahead the last vehicle, this vehicle becomes a traditional truck while the others keep on driving as a platoon.

3.7. *AdaptHdwy*

AdaptHdwy function takes in input the points along the network where the platoons become aware of the ramps and start adapting their headway (Figure 6); said points are defined through the *RampAhead* function before the simulation starts (not reported in the paper for readability but it may be shared by the corresponding author upon request). This function is not in the original script and is added ex-novo.

The change in the headway value is obtained through a change in vehicle type for the following vehicles, for which another driving behaviour is defined through *PTV Vissim*. This behaviour should be similar to the one used in normal driving conditions but with a different CC1 parameter in the longitudinal control law. By the means of this parameter, *PTV Vissim* defines the time gap that characterizes the driving behaviour and, finally, the kept headway. The new CC1 value should range between 1 and 2 s, to allow other vehicles to cut-in through the platoon and to engage the upcoming off-ramp. It is advisable for the CC1 value to be a fixed time distribution because the platoon is considered still in formation and thus no oscillation in the headway value should arise. When this function is called, the platoon remains in formation even if an external vehicles cut-ins (a relevant difference from the cut-in function defined above implemented through the *vehicleTypeR*),

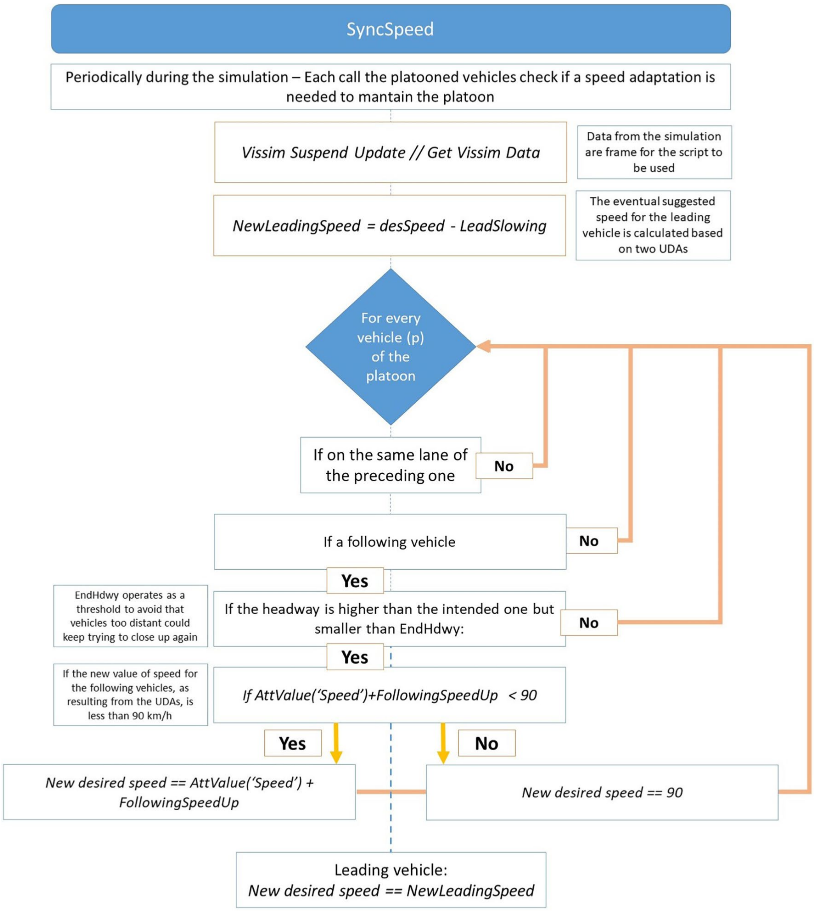


Figure 4. *SyncSpeed* function

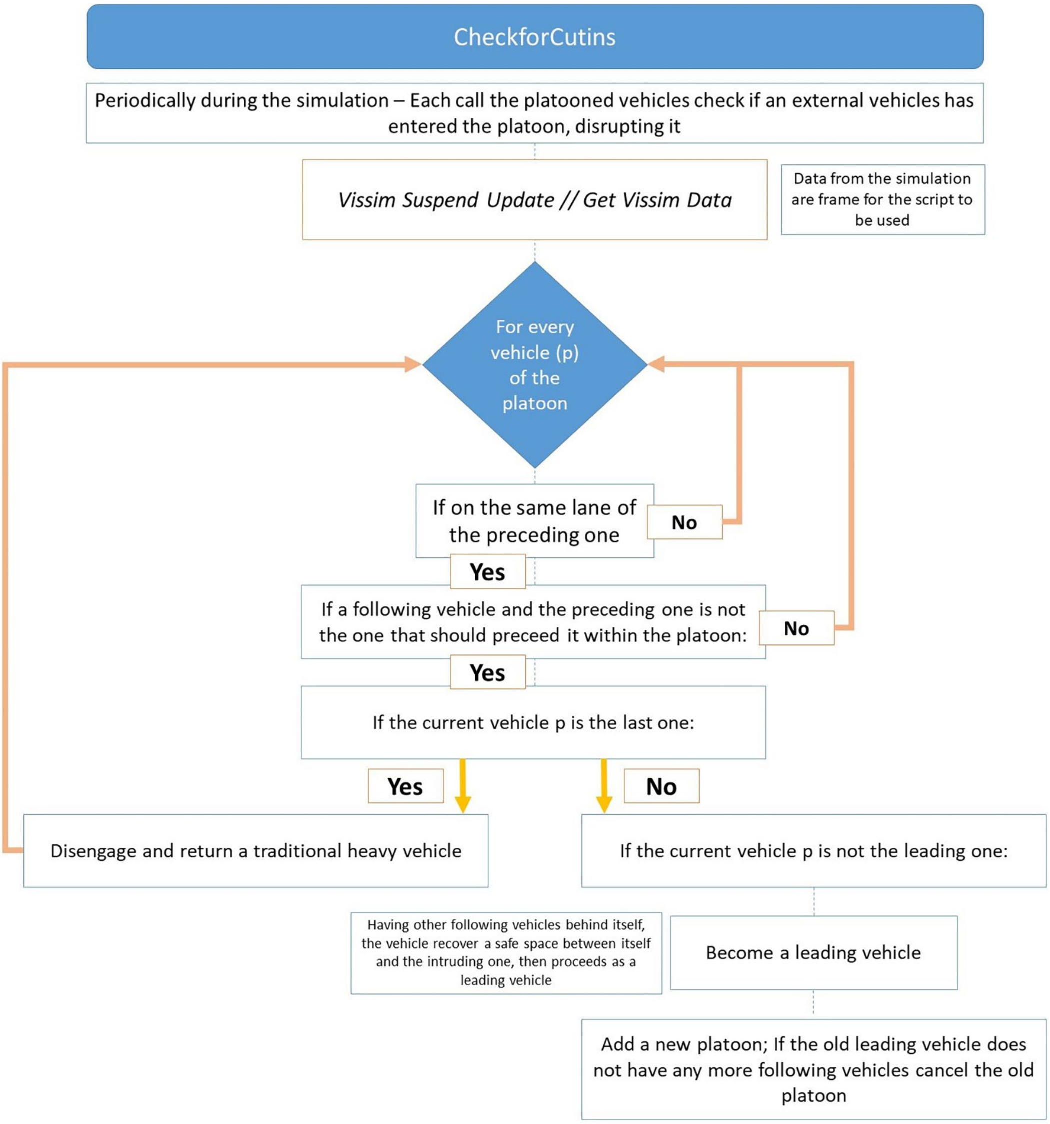


Figure 5. *CheckforCutins* function

this choice was made because these manoeuvres are carried out only near ramp areas and, therefore, the intruding vehicles don't remain for long inside the platoon. It should be highlighted, though, that the platooning vehicles stop to synchronize their speed (as it should be, due to the intruded external vehicle). As soon as the ramp is surpassed by the leading vehicle, the whole platoon returns to its original state.

4. Analysis of truck platooning at on-ramp areas – assessment of different strategies

This section faces the second aim of the paper, the analysis of three different platooning strategies and their impact at on-ramp areas. These three strategies are: *driving as usual*, *headway adaptation* and platoon *dissolution*. In fact, the main issue investigated in this section is how the merging traffic would be impacted by a certain share of platoons of trucks on the main branch. It is acknowledged in literature, as reported in Section 1, that the strongly reduced headways among platooning vehicles can hinder the merging manoeuvre and decrease the merging speed of the vehicles entering through the on-ramp (it should be highlighted that within the paper, merging does not refer to the manoeuvre of trucks adding themselves to the platoons but refers to vehicles entering the main branch through the on-ramp). Therefore, based on the know-how developed within the C-Roads Italy project and described by Agriesti *et al.* (2018b), the three possible strategies were defined as in the following and scripted within the tool presented in Section 3:

- *driving as usual*: following this strategy, the platoons of trucks do not increase the headway and keep on driving as close as possible, to minimize the occupied space and reduce the hindering effect on the on-ramp;
- *headway adaptation*: following this strategy, the platoons of trucks increase the kept headway to potentially allow merging vehicle to exploit this space and enter the main branch (Section 3.7);

- *dissolution*: following this strategy, the platoons of trucks dissolve before arriving to the on-ramp area, nullifying the hindering effect on the merging traffic.

The simulation layout is reported in Figure 7 (where the purple vehicles are the truck platooning ones and are limited to the first lane).

It is important to note that, even though the traffic data used as input had vehicular composition similar to the one exploited within the C-Roads Italy project (Agriesti *et al.* 2020), the simulated framework does not refer to a specific infrastructure or traffic and, therefore, the calibration process was based on the *Highway Capacity Manual* (TRB 2010) with the aim of reproducing the actual capacity constraint of an on-ramp segment with two lanes on the main branch. The full process is reported by Aleccia (2019), work on which this analysis is based, in Figure 8 the final results obtained for the calibration are showed (namely the traffic volume degrading the speed down to 80 km/h).

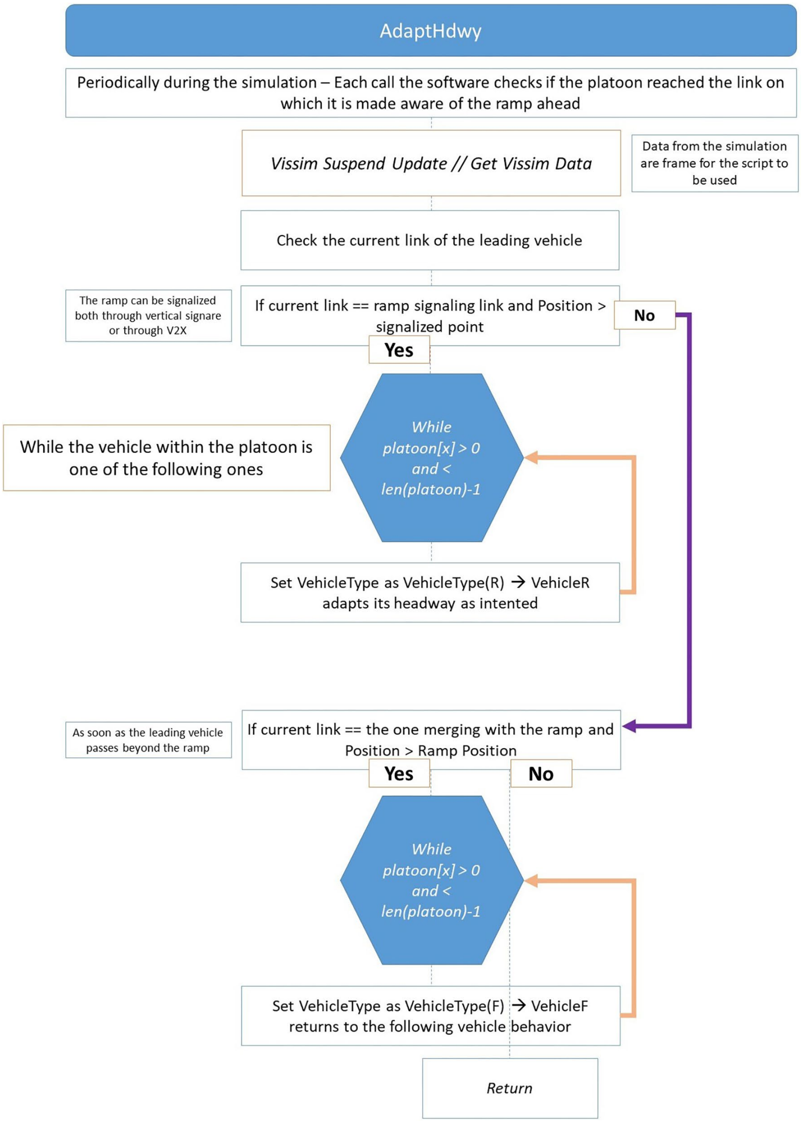


Figure 6. *AdaptHdwy* function

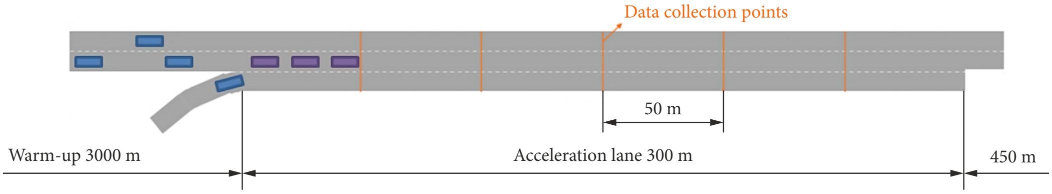


Figure 7. Modelling layout – adapted from research by Aleccia (2019)

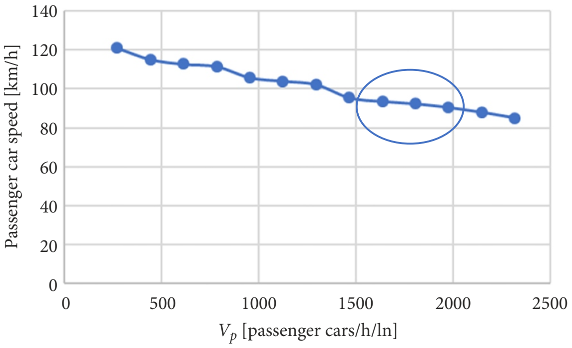


Figure 8. Modelled output – capacity of the on-ramp segment

Different MPRs for the truck platooning system were simulated: 0, 20, 40 and 60% (Table 2); the increasing of the MPR reduces the number of traditional heavy vehicles within the traffic flow. Moreover, two headway values (20 and 30 m) were tested through micro-simulations.

The micro-simulations were performed through 6 multi-run for each scenario, as by best practices, by increasing the random seed for each run. The results that were obtained with a headway equal to 20 m are presented in the following, while the ones concerning the 30 m gap are not reported, following the same trends that were obtained in 20 m and only changing in absolute values. Please refer to research by Aleccia (2019) for the whole set of results.

4.1. 20 m headway – driving as usual

The first KPI chosen was the average speed across the segment, visualized in Figure 9; this KPI should frame both the possible impacts of a platoon dissolving or adapting its headway and also the impact that the merging traffic and its speed can have on the main branch for the different strategies.

It is interesting to note how a small percentage of platoons of trucks decreases the average speed while higher MPRs entail a benefit for the overall traffic. This effect is mostly in line with what can be found in research by Agriesti *et al.* (2018b) and is likely due to how vehicles arrange themselves on the two lanes: for few platoons the other vehicles do overtake and re-enter the first lane more often, increasing the disrupting effects, while for higher values less vehicles re-enter the first lane and keep on driving on the fast one. It should also be pointed out how being in a platoon forces the heavy vehicles to better comply to the speed limits, which in turn decreases the mean speed of the heavy vehicles on the main branch.

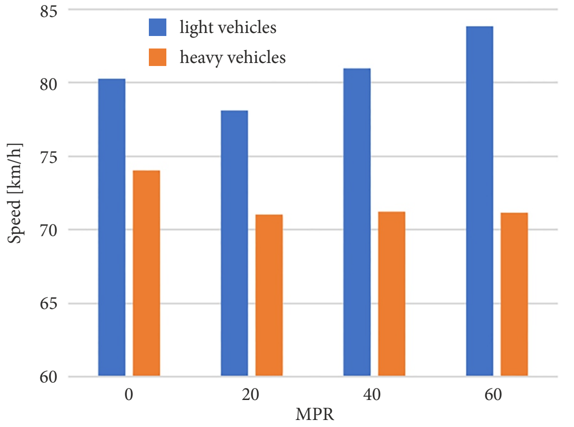


Figure 9. Speed of light vehicles (blue) and of heavy vehicles (orange) for different MPRs [km/h] – *driving as usual* strategy

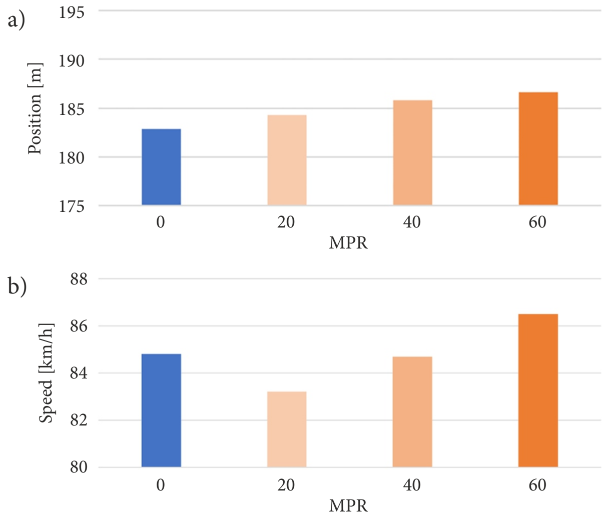


Figure 10. Merging position on the acceleration lane [m] (a) and average speed on the acceleration lane [km/h] (b) – *driving as usual* strategy

Moreover, the effect of the *driving as usual* strategy on the merging vehicles is framed through the position of merging of said vehicles, as reported in Figure 10.

This result is coherent with the research hypothesis stating that an increased number of platoons on the main branch actually hinders the merging manoeuvres, delaying them. It is worth highlighting that vehicles on the fast lane tend to accelerate to merge ahead the platoon rather than behind, the safer option among the two because minimizes the speed difference between on-ramp and main traffic flows.

Table 2. Model input for different MPR

MPR [%]	Total flow	# light vehicles	# heavy vehicles	% light vehicles	% heavy vehicles	# platoons	Main flow	On-ramp flow
0	2400…2800	1980…2380	420	83…85	17…15	0	1920…2240	480…560
20	2400…2800	1980…2380	336	85…88	15…12	28	1853…2173	463…543
40	2400…2800	1980…2380	252	89…90	11…10	56	1786…2106	446…526
60	2400…2800	1980…2380	168	92…93	8…7	84	1718…2038	430..510

4.2. 20 m headway – headway adaptation

The research hypothesis for this strategy is that, by increasing the gap up to the traditional values (the ones kept by the traditional heavy vehicles), merging vehicles could exploit the additional space to merge as they please (Figure 11).

This time, the change from a negative to a positive trend arises later, for MPR of 60% or higher. This had to be expected because in adapting itself, the platoon imposes decelerations on its following vehicles (to recover a traditional headway) and consequently deceleration on the other vehicles on the slow lane. Still, the presence of platooning has entails benefits for the main stream that surpass the negative impacts for MPR higher than 40% – this results is coherent with the ones found in research by Agriesti *et al.* (2018b).

From the results, it appears how restoring a traditional headway does not foster vehicles on the merging lane in their manoeuvre; nevertheless, it is hardly conceivable to increase the headway to values higher than the ones performed by traditional heavy vehicles without risking implementing the *dissolution* strategy instead. Therefore, a relevant output of this set of simulation is that the headway adaptation does not perform better than the *driving as usual* strategy; it performs worse instead due to the imposed decelerations, as explained above. It should be highlighted, though, how the merging aggressiveness of the merging vehicles is calibrated to reproduce the same capacity at on-ramp areas as the one identified within the *Highway Capacity Manual* (TRB 2010). Driving aggressiveness towards platooning is a parameter that is still lacking in literature due to truck platooning being in prototypal stage, therefore as a future research direction the aggressiveness of merging vehicles should be validated through field-tests such as the ones that will be performed within C-Roads Italy during 2020. Merging position and average speed on the acceleration lane according to headway adaptation strategy visualized in Figure 12.

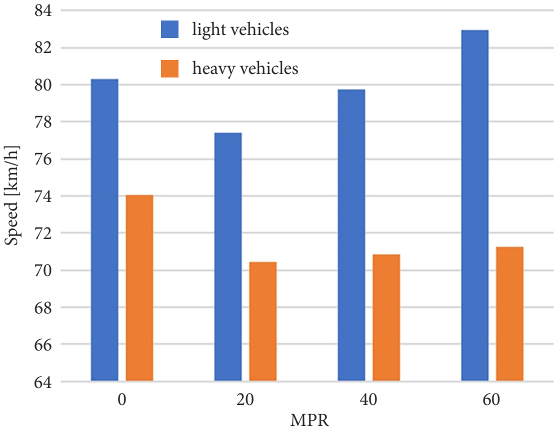


Figure 11. Speed of light vehicles (blue) and of heavy vehicles (orange) for different MPRs [km/h] – *headway adaptation* strategy

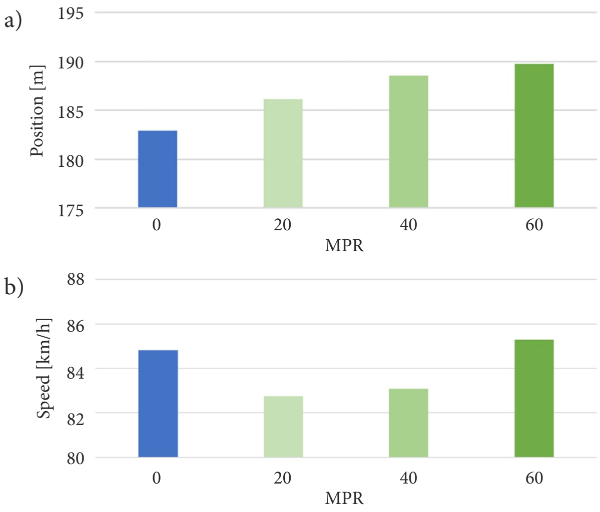


Figure 12. Merging position on the acceleration lane [m] (a) and average speed on the acceleration lane [km/h] (b) – *headway adaptation* strategy

4.3. 20 m headway – dissolution

This last strategy has the following vehicles disengaging from the leading one, upstream the on-ramp so that the heavy vehicles involved in a platoon arrive to the on-ramp as traditional vehicles, which means that they not only recover traditional headways but can also overtake and they have more space to mix and adapt to the surrounding traffic flow. The script implements this strategy in a way that nullifies the effects of the platoons at the on-ramp areas, meaning that the traffic in these sections is completely comparable with the 0% MPR. The obtained results are presented in Figure 13.

It is relevant to note how, by implementing this strategy, the average speed of light vehicles reaches values that are really close to the *driving as usual* strategy (for 20 MPR, for example, the loss in average speed is equal to 2.19 km/h in *driving as usual* and to 2.68 km/h in *dissolution* while for 60 MPR the gains are 2.13 and 2.73 km/h, respectively). The slightly higher benefit in the *dissolution* strategy are due to heavy vehicles resuming the traditional driving speed, not perfectly compliant with the speed limit of 80 km/h. Still it should not be forgotten that in the *dissolution* strategy all the benefits related to truck platooning are lost. The similar impact of *driving as usual* and *dissolution* over speed may be explained considering the limited number of heavy vehicles within the traffic flow that limits the different impacts of the strategies. As a future research direction, a higher number of scenarios with different traffic flows and composition should be analysed through the script to widen the set of results and their worth. Still, a valuable result arises from the *dissolution* strategy for the merging vehicles on the acceleration lane (Figure 14).

It appears how for the average merging position on the acceleration lane and the average merging speed, the results are in line with the *driving as usual* strategy, even if slightly worse in terms of traffic efficiency. This is relevant because it shows how, even if the platoon does dissolve in advance, the clustering of heavy vehicles arriving at the on-ramp quite close to each other delays the merging of the on-ramp vehicles (compared to the *no platooning* results in which heavy vehicles get to the on-ramp scattered). Moreover, the *driving as usual* strategy outperforms the others likely due to the merging vehicles accelerating to overtake the platoon rather than slowing down to let it pass. Still, this entails one risk that would not be considered if not for the results in the Figure 15 – the *dissolution* strategy is the only one that limits the number of merging vehicles that end up stopping at the end of the acceleration lane.

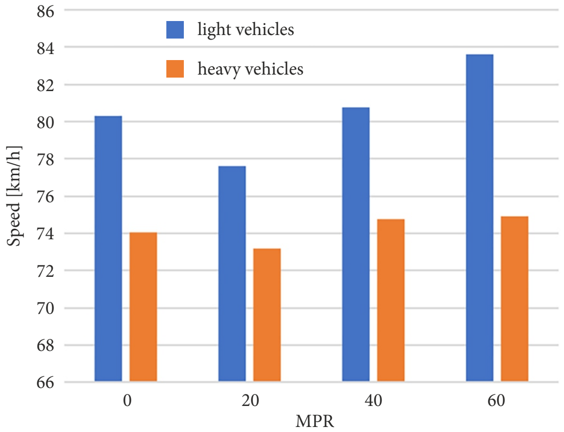


Figure 13. Speed of light vehicles (blue) and of heavy vehicles (orange) for different MPRs [km/h] – *dissolution* strategy

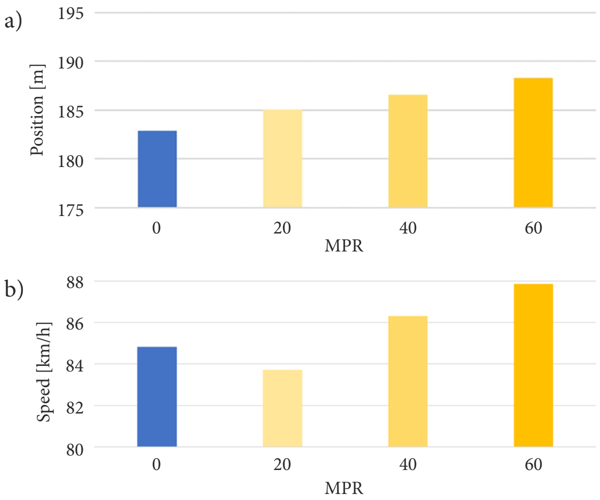


Figure 14. Merging position on the acceleration lane [m] (a) and average speed on the acceleration lane [km/h] (b) – *dissolution* strategy

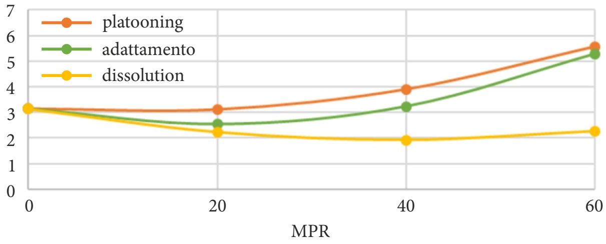


Figure 15. Failed merging attempt [% of vehicles failing to merge over the flow entering through the on-ramp] – *driving as usual* (orange); *headway adaptation* (green); *dissolution* (yellow)

Table 3. Overview of the numerical outputs concerning merging vehicles

Strategy	% vehicles failing to merge	Merging position [m] (average)	Merging speed [km/h] (average)
No platooning	3.1	183	84.8
Driving as usual	6.0	188	85.2
Headway adaptation	5.4	192	83.2
Dissolution	1.9	185	89.8

This result should not be overlooked: even though the KPI does not directly translates into an impact on traffic efficiency, it is quite relevant on safety. It is clear how for every vehicle that does not find a gap to merge and stops on the acceleration lane, the risk of collision when actually merging is way higher than acceptable (Table 3). This would suggest that between the three driving strategies, it is actually *dissolution* that performs better, scoring results similar to *driving as usual* on the main branch but eliminating many of the risky situations from the on-ramp. Moreover, the slight better performance in Figures 11 and 12 of the *headway adaptation* strategy when compared with *driving as usual* strategy validates the functioning of the script, namely merging vehicles do actually consider the increased space within the platooning vehicles and if really needed perform a cut-in to merge with the main traffic flow. The low impact of the *headway adaptation* strategy is therefore due to a low willingness of the on-ramp vehicles to take advantage of the increased space within heavy vehicles. In the following table, an overview of the three strategies is reported: as it can be seen *dissolution* performs better than the other two in addressing the issue of merging vehicles, while *driving as usual* foster earlier and speedier merging than *headway adaptation*, but entails higher risks of merging failures among the on-ramp vehicles.

Conclusions and future works

As reported in the Section 1, there is an immediate need of a more comprehensive literature facing the subject of traffic simulations assessing the impacts of truck platooning and at designing field-tests in a scientific way. The issue commonly faced by research bodies is the lack of a common tool that would replicate the behaviour of a platoon of trucks in a simple and flexible way. To provide this tool would allow evaluators across the European countries to study the same system, with the same behaviour, through the definition of a small set of parameters to reproduce the boundary conditions better representing the national realities. This paper tries to provide such a tool, integrated at least in one of the most common traffic simulation software on the market. The presented script was designed based on a sound theoretical background and to be simple enough to allow even the ones with no skills in programming to reproduce the desired truck platooning behaviour.

The script as reported was then exploited to obtain a first assessment useful both to widen the results currently present in literature and to obtain a first assessment useful for the activities of the C-Roads Italy project. Aim of the paper was, in fact, to define the best strategy to minimize the negative impacts of the truck platooning at on-ramp areas, an issue well acknowledged in literature. Three strategies were simulated through the script: *driving as usual*, *headway adaptation* and *dissolution*, designed following best practices and implementation logics as found in literature and implemented with a high level of detail thanks to the script (as reported in the first half of the paper). The results showed that both *driving as usual* and *dissolution* perform better than *headway adaptation* on the main branch, due to the low propensity of the merging vehicles to cut-in even through a platoon with traditional spacing between vehicles. A relevant research direction that emerged is the need of validate this propensity through field-tests or virtual simulations. A second set of results concerned the merging vehicles, for which the dissolution strategy seems to entail the highest benefits both in terms of average speed and in terms of safety for MPRs higher than 40%. *Headway adaptation* strategy performs better than *driving as usual* on the safety impact area but overall lowers the average speed of merging vehicles, thus compromising traffic efficiency.

Future research directions would involve the application of the scripted tool to evaluate a higher number of scenarios, both in terms of traffic composition and in terms of infrastructural junctions that could prove challenging for truck platooning. Within the activities of C-Roads Italy, studies will be carried out at roadworks also analysing the possible benefits arising from the jointed implementation of truck platooning and Cooperative Intelligent Transport System (C-ITS).

Appendix

```
def DissolvePlatoon1():
    Vissim.SuspendUpdateGUI();
    GetVissimData();
    list = []    # An empty list is created, in this list the platoons to be removed from the list of all platoons
                 # (due to the dissolution) will be inserted
    for platoon in platoons:
        LeadID = platoon[len(platoon)-1]
        LeadLink = Vissim.Net.Vehicles.ItemByKey(LeadID).AttValue(‘Lane\Link\No’)
        if LeadLink == LinkOfReception:   # If the leading vehicle enters the first link within the range of a
                                            cooperative message, the whole platoon receives the message
                                          # and starts the dissolution
            for p in platoon:
                DissolutionStart[p] = Vissim.Net.Vehicles.ItemByKey(p).AttValue(‘SimSec’)
                # DissolutionStart records the start of the dissolution maneuver that should be terminated
                # after a certain time interval
                if Vissim.Net.Vehicles.ItemByKey(p).AttValue(‘vehType’) == str(vehicleTypeL):
                    Vissim.Net.Vehicles.ItemByKey(p).SetAttValue(‘vehType’, str(vehicleType))
                elif Vissim.Net.Vehicles.ItemByKey(p).AttValue(‘vehType’) == str(vehicleTypeF):
                    DissolvedList.append(p)
                    Vissim.Net.Vehicles.ItemByKey(p):SetAttValue(‘vehType’, str(vehTypeD))
            list.append(platoon)
    for x in list:
        if x in platoons:
            platoons.remove(x)
    Vissim.ResumeUpdateGUI();

def DissolvePlatoon2():
    Vissim.SuspendUpdateGUI();
    GetVissimData();
    List = []    # An empty list is created, in this list the vehicles to be removed from the platoon will be
                 # inserted
    for ID in DissolvedList:   # Through DissolvedList, the function checks what vehicles are dissolving and
                               # if the dissolution maneuver can come to an end
    if Vissim.Net.Vehicles.ItemByKey(ID).AttValue(‘Speed’) > 50:
    # For uncongested traffic conditions, the vehicles complete the dissolution maneuver when they reac
    # quire a traditional headway (or if the vehicle ahead changes lane)
    if Vissim.Net.Vehicles.ItemByKey(ID).AttValue(‘VehType’) == str(vehicleTypeD) and
    ↳(float(Vissim.Net.Vehicles.ItemByKey(ID).AttValue(‘Hdwy’)) >= DissHdwy or
    ↳float(Vissim.Net.Vehicles.ItemByKey(ID).AttValue(‘Hdwy’)) == 0):
        Vissim.Net.Vehicles.ItemByKey(ID).SetAttValue(‘vehType’, str(vehicleType))
        DissolvedList.remove(ID)
        List.append(ID)
    else:
    # For congested traffic conditions it is the time that rules over the end of the dissolution maneuver
     if Vissim.Net.Vehicles.ItemByKey(ID).AttValue(‘vehType’) == str(vehicleTypeD) and
     ↳Vissim.Net.Vehicles.ItemByKey(ID).AttValue(‘SimSec’) > float(DissolutionStart[ID])+MaxTimeTransition:
        Vissim.Net.Vehicles.ItemByKey(ID).SetAttValue(‘VehType’, str(vehicleType))
        DissolvedList.remove(ID)
        List.append(ID)
  for y in List:
    if y in DissolvedList:
        DissolvedList.remove(y)
```

References

Agriesti, S.; Gandini, P.; Marchionni, G.; Paglino, V.; Ponti, M.; Studer, L. 2018a. Evaluation approach for a combined implementation of day 1 C-ITS and truck platooning, in *2018 IEEE 87th Vehicular Technology Conference* (*VTC Spring*), 3–6 June 2018, Porto, Portugal, 1–6. https://doi.org/10.1109/VTCSpring.2018.8417876

Agriesti, S.; Studer, L.; Ponti, M.; Gandini, P.; Marchionni, G.; Maja, R.; Borghetti, F.; Paglino, V.; Bruglieri, M. 2018b. *Evaluation Approach for a Combined Implementation of Day 1 C-ITS and Truck Platooning – v.1.0*. Ex-Ante Evaluation Report. C-Roads Italy. 232 p.

Agriesti, S.; Studer, L.; Marchionni, G.; Gandini, P.; Qu, X. 2020. Roadworks warning – closure of a lane, the impact of C-ITS messages, *Infrastructures* 5(3): 27. https://doi.org/10.3390/infrastructures5030027

Aleccia, O. 2019. *Impatti del truck platooning sul traffico: il caso studio delle rampe di ingresso*. Tesi di laurea Magistrale. Politecnico di Milano, Repubblica Italiana. Available from Internet: https://www.politesi.polimi.it/handle/10589/150770 (in Italian).

Andersson, J.; Englund, C.; Voronov, A. 2017. *Study of Communication Needs in Interaction between Trucks and Surrounding Traffic in Platooning*. Drive Sweden, Lindholmen Science Park, Gothenburg, Sweden. 22 p. Available from Internet: https://www.drivesweden.net/sites/default/files/content/final_report_-_40016_submitted.pdf

Bakermans, B. A. 2016. *Truck Platooning: Enablers, Barriers, Potential and Impact*. MSc Thesis, Delft University of Technology, The Netherlands. 85 p. Available from Internet: https://repository.tudelft.nl/islandora/object/uuid:3e4e265b-b84c-474b-85b9-9840a5c85464

Benza, M.; Bersani, C.; D'Incà, M.; Roncoli, C.; Sacile, R.; Trotta, A.; Pizzorni, D.; Briata, S.; Ridolfi, R. 2012. Intelligent transport systems (ITS) applications on dangerous good transport on road in Italy, in *2012 7th International Conference on System of Systems Engineering* (*SoSE*), 16–19 July 2012, Genova, Italy, 223–228. https://doi.org/10.1109/SYSoSE.2012.6384180

Bishop, R. 2017. Connected automated trucking: latest developments and outlook, in *5th Annual Florida Automated Vehicles* (*FAV*) *Summit*, 14–15 November 2017, Tampa, FL, US. 34 p. Available from Internet: https://favsummit.com/pdfs/richard-bishop.pdf

Brizzolara, D.; Toth, A. 2016. The emergence of truck platooning, *Baltic Transport Journal* (2): 58–59.

Deng, Q. 2016. *Heavy-Duty Vehicle Platooning: Modeling and Analysis*. Licentiate Thesis. KTH Royal Institute of Technology, Stockholm, Sweden. 64 p. Available from Internet: http://kth.diva-portal.org/smash/record.jsf?pid=diva2%3A901974&dswid=433

EC. 2018. *Communication from the Commission to the European Parliament, the Council, the European Economic and Social Committee, the Committee of the Regions on the Road to Automated Mobility: an EU Strategy for Mobility of the Future*. COM/2018/283 Final, 17 May 2018, Brussels, Belgium. 18 p. Available from Internet: https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX%3A52018DC0283

ERTRAC. 2017. *Automated Driving Roadmap*. Version 7.0. European Road Transport Research Advisory Council (ERTRAC) Working Group "Connectivity and Automated Driving". 56 p. Available from Internet: https://www.ertrac.org/uploads/documentsearch/id48/ERTRAC_Automated_Driving_2017.pdf

Gordon, M. M. 2015. *Evaluation of Driver Assistive Truck Platooning on Traffic Flow*. MSc Thesis. Auburn University, AL, US. 65 p. Available from Internet: https://etd.auburn.edu/bitstream/handle/10415/4693/MMG_Thesis.pdf

Lammert, M.; Duran, A.; Diez, J.; Burton K.; Nicholson, A. 2014. Effect of platooning on fuel consumption of class 8 vehicles over a range of speeds, Following Distances, and Mass, *SAE International Journal of Commercial Vehicles* 7(2): 626–639. https://doi.org/10.4271/2014-01-2438

Liang, K.-Y.; Mårtensson, J.; Johansson, K. H. 2014. Fuel-saving potentials of platooning evaluated through sparse heavy-duty vehicle position data, in *2014 IEEE Intelligent Vehicles Symposium Proceedings*, 8–11 June 2014, Dearborn, MI, US, 1061–1068. https://doi.org/10.1109/IVS.2014.6856540

Ploeg, J.; Serrarens A. F. A.; Heijenk, G. J. 2011. Connect & drive: design and evaluation of cooperative adaptive cruise control for congestion reduction, *Journal of Modern Transportation* 19(3): 207–213. https://doi.org/10.1007/BF03325760

Ramezani, H.; Shladover, S. E.; Lu, X.-Y.; Chou, F.-C. 2018. *Microsimulation Framework to Explore Impact of Truck Platooning on Traffic Operation and Energy Consumption: Development and Case Study*. PATH Research Report for FHWA Exploratory Advanced Research Program Cooperative Agreement DTFH61-13-H00012 Tasks 1.4 and 2.7. Institute of Transportation Studies, University of California, Berkeley, CA, US. 54 p. Available from Internet: https://path.berkeley.edu/sites/default/files/fhwa_report_-_2-27-2018_.pdf

Ricardo UK Ltd. 2014. *Heavy Vehicle Platoons on UK Roads: Feasibility Study*. Document No RD.14/57101.2. Ricardo UK Ltd., UK. 107 p. Available from Internet: https://assets.publishing.service.gov.uk/government/uploads/system/uploads/attachment_data/file/637361/truck-platooning-uk-feasibility-study.pdf

SAE International. 2016. *Taxonomy and Definitions for Terms Related to Driving Automation Systems for On-Road Motor Vehicles*. Ground Vehicle Standard J3016_201609. https://doi.org/10.4271/J3016_201609

Sia Partners. 2016. *Platooning Towards Sustainable Road Freight Transport: the Benefits, Drawbacks and Future Outlook*. Sia Partners, Paris, France. 9 p. Available from Internet: https://www.sia-partners.com/system/files/document_download/file/2020-06/insight_platooning_-_benefits_drawbacks_and_future_outlook.pdf

Smith, D. A. 2016. *Modeling and Understanding the Implications of Future Truck Technology Scenarios for Performance-Based Freight Corridor Planning*. PhD Dissertation. Georgia Institute of Technology, GA, US. 134 p. Available from Internet: https://smartech.gatech.edu/handle/1853/56245

Studer, L.; Agriesti, S.; Gandini, P.; Marchionni, G.; Ponti, M. 2019. Impact assessment of cooperative and automated vehicles, in M. Lu (Ed.). *Cooperative Intelligent Transport Systems: Towards High-Level Automated Driving*, 397–417. https://doi.org/10.1049/PBTR025E_ch18

Studer, L.; Frötscher, A.; Crockford, G.; Gandini, P.; Marchionni, G.; Netten, B.; Vlassenroot, S.; Kwakernaat, H. J.; Zimmermann, W.; Wadsworth, P.; Gutiérrez Lanza, S.; Innamaa, S.; Schirokoff, A.; Dörge, L.; Carvalho, D.; Tegenbos, R.; Chalhoub, G.; Geißler, T.; Merlin, B.; Nelen, L.; Vanis, M.; Herrero Zarzosa, J. I.; Nagy, Á.; Aniss, H.; Esposito, M. C.; Burianne, A. 2018. *Evaluation and Assessment Plan*. Final Version. Working Group 3, C-Roads Platform, Austria. 71 p. Available from Internet: https://www.c-roads.eu/fileadmin/user_upload/media/Dokumente/C-Roads_WG3_Evaluation_and_Assessment_Plan_Final.pdf

TRB. 2010. *Highway Capacity Manual*. Transportation Research Board (TRB), Washington, DC, US. 1650 p.

Van Maarseveen, S. 2017. *Impacts of Truck Platooning at Motorway On-ramps*. MSc Thesis. Delft University of Technology, The Netherlands. 174 p. Available from Internet: https://repository.tudelft.nl/islandora/object/uuid:dd8ee414-bddc-435b-90fe-21480cbd9187