



Article scientifique

Article

2025

Published version

Public access

This is the published version of the publication, made available in accordance with the publisher's policy.

---

## Quantitative global analysis of a lane merging strategy for collaborative autonomous vehicles

---

Chopard, Bastien; Leone, Pierre; Lukic, Luka

### How to cite

CHOPARD, Bastien, LEONE, Pierre, LUKIC, Luka. Quantitative global analysis of a lane merging strategy for collaborative autonomous vehicles. In: Natural computing, 2025, vol. 24, p. 879–893. doi: 10.1007/s11047-025-10049-2

This publication URL: <https://archive-ouverte.unige.ch/unige:189560>

Publication DOI: [10.1007/s11047-025-10049-2](https://doi.org/10.1007/s11047-025-10049-2)

© The author(s). This work is licensed under a Creative Commons Attribution (CC BY 4.0)

<https://creativecommons.org/licenses/by/4.0>

Last deposit update in Archive ouverte UNIGE on 10.12.2025 14:26



# Quantitative global analysis of a lane merging strategy for collaborative autonomous vehicles

Bastien Chopard<sup>1</sup> · Pierre Leone<sup>1</sup> · Luka Lukic<sup>1</sup>

Accepted: 27 July 2025 / Published online: 19 August 2025  
© The Author(s) 2025

## Abstract

In this paper we suggest a simple algorithm for merging the traffic of a main road with on-ramp traffic for collaborative Connected and Autonomous Vehicles (CAVs). The main asset of the algorithm being that most values can be formally computed, we quantitatively express the global impact of the merging strategy on the input and output flows of the merging area. For instance, how platoons emerge, their lengths, the space between them and their traffic patterns. We refer to the literature for all local aspects of synchronisation of cars. Here we simply assume that cars can execute the merging algorithm which requires that the cars are aware of a sequence number and operates following the sequence. Our analysis focuses on saturated flows and the control of collective behavior.

**Keywords** Lane merging · Autonomous vehicles · Kinematic waves · Maximal traffic flow · Platoons formation · Collective behavior

## 1 Introduction

A classical problem in traffic control is the lane merging problem where there is a infinite traffic pattern on a mainstream lane that has to be merged with some exogenous traffic coming from an on-ramp, see Fig. 1. All of us know this configuration as a potential bottleneck leading mainstream cars to stop.

This paper deals with this problem in the context of Autonomous and Connected Vehicles (CAVs). The merging strategy can then make use of some knowledge about the traffic patterns and coordinate the vehicles by exchanging messages. We do not discuss the communication protocols here, we refer to Zhu et al. (2022) and references therein as well as some other works cited in Section 2.

Autonomous cars are driven by a *regulator* which gathers information (head-tail distance to the next car, car on the side, speeds, etc.) and sets the speed of the car accordingly. In this work we assume the regulator functioning in two modes: 1. Car-following and, 2. Merging.

The merging mode is the subject of this paper. We present a simple algorithm to merge two lines of cars. We do not claim that the algorithm is optimal but it has the great quality of leading to close form expressions for the values of interest. Besides computing the time needed to merge two lines of cars we are interested to understand the impact of the merging on the traffic pattern and in particular how often it can be repeated (to compute a flow of cars on-ramp). We also consider the impact of an on-ramp on previous and successive on-ramp. Merging two lines of cars generate traffic waves that may impede merging at some distant points of the road.

The car following algorithm that we consider is supposedly among the simplest ones (Newell 2002). We assume that the flows of cars on the mainstream and on-ramp are initially *saturated* meaning that the head-tail distances between cars is such that a car copies the action of the next car (when the next car decelerates/accelerates at time  $t$  it decelerates/accelerates at time  $t + 1$ ), see (Cohen et al. 2022, 2023) for a discussion of how such pattern emerges.

---

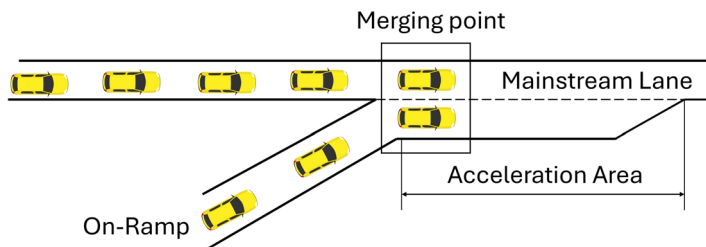
✉ Pierre Leone  
pierre.leone@unige.ch

Bastien Chopard  
Bastien.Chopard@unige.ch

Luka Lukic  
Luka.Lukic@etu.unige.ch

<sup>1</sup> Computer Science Department, University of Geneva, Geneva, Switzerland

**Fig. 1** Illustration of the lane merging problem at on-ramp



**Fig. 2** Three platoons, the leader of each platoon can accelerate since the head-tail distance with the last trailing car of the next platoon is large enough, i.e. its speed is free

Generally, a saturated flow is distinguished by the fact that there is a car *leader* in front of the line of cars whose speed is free and the following cars called *trailing* cars whose speeds depend on the speed of their leading cars. Formally, if the *leader* car is 0, the following ones 1, 2, . . . we denote  $v^n(t)$  the speed of car  $n$  at time  $t$  it holds that  $v^0(t)$  is free and

$$v^n(t) = v^{n-1}(t - 1). \tag{1}$$

The group composed of the leader and the trailing cars is called a *platoon* (Charlottin et al. 2024), see Fig. 2. A trailing car can become a leader if its speed decreases below the speed of its leading car. Reciprocally, a leader car can become a trailing car if it joins the last car of the next platoon, i.e. two platoons merge.

Notice that only a platoon leader can start the merging algorithm and it is needed that it can accelerate sufficiently without merging with the next platoon.

The traffic from the on-ramp comes by group of  $N/2$  cars ( $N$  always assumed even) with a similar traffic pattern as on the mainstream lane.

The global picture of the merging strategy to include a new group of  $N/2$  cars requires first that the cars on the mainstream decelerate from the cruise speed  $v_{max}$  to speed  $v$  when arriving to the merging area. Then, on-ramp traffic is merged and finally cars accelerate to the cruise speed  $v_{max}$ , see Fig. 3.

## 2 Related works

In this section we refer only to merging strategies with CAVs. In particular we do not consider works that try to understand the real traffic patterns due to the human drivers, or make statistical analysis due to the lack of control on the traffic patterns, see for instance (Sicheng Sun et al. 2021; Guzmán et al. 2015).

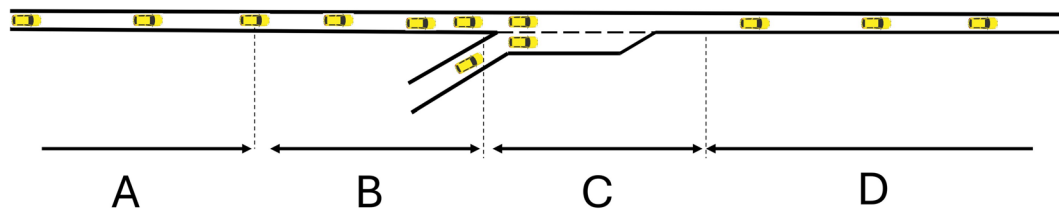
One important assumption that we make is that the cars have the ability to adopt a given traffic pattern. For the cars

in the main road this means that they move at the same speed  $v$  and head-tail distance  $d = v$ . This follows from the car-following speed regulator of Cohen et al. (2022, 2023) with the possibility of managing precisely the head-tail distance with communications. The same traffic pattern is also required for the incoming cars from the on-ramp.

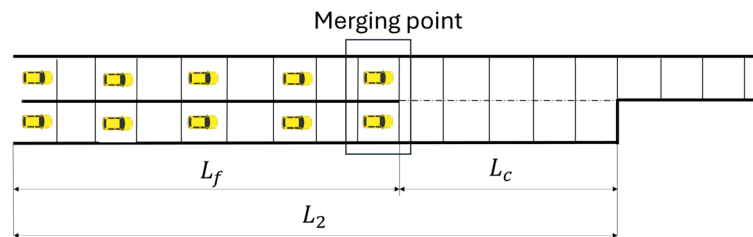
The recent review (Zhu et al. 2022) presents a relevant panel of strategies for the coordination of CAVs. We discuss some strategies here with the aim of making clear how our approach is different. In this work we focus on the design of a speed regulator leading to a vehicle based model of the traffic We aim at the design of a regulator that takes as input the road conditions and output the action a car follows and at the analysis of the traffic dynamics induced by the regulator. An alternative is to model traffic with continuous model and expressing the dynamics as differential equations with constraint laws (Rios-Torres and Malikopoulos 2016; Newell 1993; Lighthill and Whitham 1955; Daganzo 2006; Bayen et al. 2022; LeVeque and Leveque 1992).

A clever strategy is proposed in Milanés et al. (2011), Huang et al. (2019) to turn a car-following regulator to a solution of the lane merging problem, *the virtual car following*. The idea is to map the cars on the other lane to the current lane and let the car-following regulator regulate the distances. Once the distances are large enough merging occurs safely. In our approach we leverage communications between cars to shorten the safety distances. Indeed, we assume that a trailing car is aware of the evolution of the leading car speed and the reaction time is reduced to 0. The regulator, that control the car’s motion, makes use of this only during the merging process.

Another important concept in the literature is the *merging sequence*. Given the traffic on the two lanes it amounts to choosing the order of the cars on the final single lane. In Linghui et al. (2019) the authors suggest to use a genetic algorithm to compute the optimal merging sequence given a traffic configuration with a complex fitness function. In



**Fig. 3** The different road sections. A: cars cruise at speed  $v_{max}$ . B: cars decelerate to speed  $v$ . C: cars apply the merging algorithm and accelerate. D: cars cruise at speed  $v_{max}$



**Fig. 4** The lane merging problem. Two lanes have to merge. Up is the target lane. In the initial conditions, cars are assumed to move at the same speed with head-tail distance equals the speed. Such a preconditioning phase is achievable with CAVs

Ntousakis et al. (2016) the authors assume a given merging sequence and coordinate the vehicles accordingly. In Petig et al. (2018) the authors assume a very simplified model and prove optimal and approximation results.

Some papers consider successive on-ramps and off-ramps to a motorway (Chen et al. 2017; Lin et al. 2012). This is a problem we address as well in this paper. The vehicle based model we use suggests that access from different on-ramps need to be synchronized. Indeed, merging cars at an on-ramp leads to a wave of decelerating cars that propagates upstream. When this waves passes an on-ramp, it prevents the merging of cars.

In general, the optimization parameters are varied: fuel consumption, average speed in the merging area, total speed of all vehicles, passenger comfort, safety requirements (Zhu et al. 2022).

Model predictive scheme is proposed in Cao et al. (2015) for the computation of the optimal merging sequence. Here, the traffic is constraint by communication to a known pattern.

In this work we assume that cars coordinate their motions to reach a particular configuration illustrated in Fig. 4 and the merging algorithm is applied from this configuration. Actually, once the configuration is reached the regulators change the functioning mode from car following to merging (regulators take advantage of communications for synchronization). The merging sequence, see Fig. 10, describes the order of accelerating cars and the order of the cars in the final line.

Notice to finish that for many works, the problem is solved locally. This means that the focus is on the coordination of vehicles to let one pass before or behind

another. The problem amounts to avoiding collision. We refer to these works for this local synchronization and consider the problem globally. Indeed, our strong assumptions on the traffic patterns leads to global understanding of the merging strategy on the traffic at large scale.

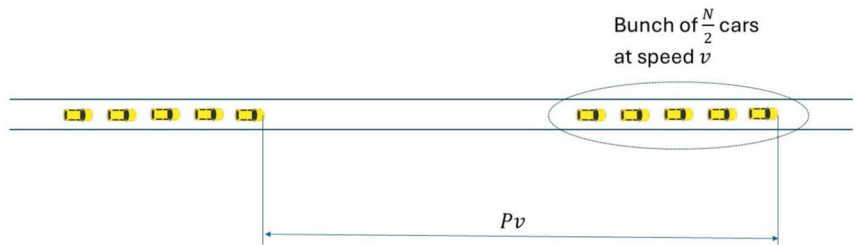
### 3 Presentation of the problem

In this article we consider the problem of On-Ramp lane merging. We assume that we have one target lane on which traffic flows and whose traffic has to be merged with incoming traffic coming from an On-Ramp. As illustrated in Fig. 4 we assume that the cars are located at discrete positions of the roads, move at discrete times  $t = 1, 2, \dots$  and occupy the two lanes by pairs.

The lane merging problem can be seen as a problem of *making room* in one lane to insert new cars. Our approach consists in making room by first decelerating the speed before the merging point and then accelerate to a maximal speed  $v_{max}$ . Actually, the speed  $v_{max}$  is the cruising speed that cars adopt before and after merging. We denote  $v$  ( $v < v_{max}$ ) the speed at which the cars arrive at the merging point.

For the road we distinguish the main road which accepts the incoming traffic and the on-ramp from which cars enter the main road, see Fig. 1. We restrict our analysis to special traffic patterns. On the main road cars moves at the same speed  $v$  equal to the head-tail distance  $d$ , i.e.  $v = d$  (this corresponds to a maximal flow traffic pattern, or what we call saturated, see Cohen et al. 2022, 2023). On-ramp

**Fig. 5** The flow on-ramp, bunches of  $\frac{N}{2}$  cars are merged with the mainstream flow with a time period  $P$



are the cars to be merged. For the merging algorithm we first consider a total of  $N$  cars. This is illustrated in Fig. 4. We analyze the merging of the  $N$  cars ( $N/2$  on the main road and  $N/2$  on-ramp) then we analyze the impact of the merging process on the traffic pattern on the main road. Usually, local strategies for lane merging are investigated and it is hard to have the impact of this local perturbation to the global traffic pattern.

Our approach is relevant because the restriction to particular traffic patterns combined with deterministic merging strategy makes possible the computation of precise quantities. For instance, the required length  $L_f$  which represents the length of the ramp is easily seen to be  $L_f = \frac{N}{2}(v + 1) - v$  where  $v$  is the speed of the car (using that for the head-tail distance  $d$  it holds that  $d = v$ ) and the car length is 1. More complex measures are (see Fig. 4 for a graphical interpretation)

- The time  $N_t$  needed before vehicles reach a maximum flow<sup>1</sup>.
- The length  $L_c$  necessary to the application of the merging lane algorithm.
- What is the achievable flow after merging? The flow being an average measure we need to describe the traffic dynamic over a long period of time to measure it. The approach we follow here is that we start by merging  $N/2$  cars with the mainstream and then compute how fast the algorithm can be repeated with another batch of  $N/2$  cars on the ramp. The achievable flow is measured as the number of merged cars by time unit.

Before presenting our algorithm and delving to the analysis we propose a rough analysis of some aspects of the problem. We consider two flows (number of cars passing per time units) entering the merging area:

- The flow on the main road which is  $\frac{v}{(v+l)}$ , cars move at speed  $v$  with head-tail distance  $d = v$ ,  $l$  is the length of the cars and we assume  $l = 1$  in the following.

- The flow on-ramp is generated by letting pass  $N/2$  cars each time period  $P$ , see Fig. 5. This means a flow of  $\frac{N}{2P}$ .

The flow out of the merging area is the sum of the two flows and is bounded by the flow of cars at maximum speed, i.e.  $v_{max}/(v_{max} + 1)$ . Hence,

$$\frac{v}{v + 1} + \frac{N}{2P} < \frac{v_{max}}{v_{max} + 1} \implies \frac{N}{2P} < \frac{v_{max} - v}{(v_{max} + 1)(v + 1)} \quad (2)$$

So we have that the on-ramp flow decreases with  $v$  indicating that a small entry speed  $v$  allows a larger on-ramp flow. This is confirmed by our analysis, see for instance the results in Fig. 19.

We described the merging problem as a problem of *making room*. This is illustrated by the following simple analysis. The  $N$  cars at speed  $v$  on the two lanes entering the merging area, the zone C in Fig. 3, span a length of  $N/2(v + 1) - v$  cells. Indeed, there are two lines of  $N/2$  cars and the head-tail distance between cars is  $d = v$ . We consider only the leader of the merging algorithm and the last car to accelerate.

The leader starts accelerating from  $v$  to  $v_{max}$  and then stays at speed  $v_{max}$  for  $x$  (to be determined) time units, covering a distance of

$$y = (v + 1) + \dots + (v_{max} - 1) + xv_{max},$$

in  $v_{max} - v - 1 + x$  time steps. During this period the last car do the same but starting by  $x$  time units at speed  $v$  before accelerating, hence covering a distance

$$z = (v + 1) + \dots + (v_{max} - 1) + xv.$$

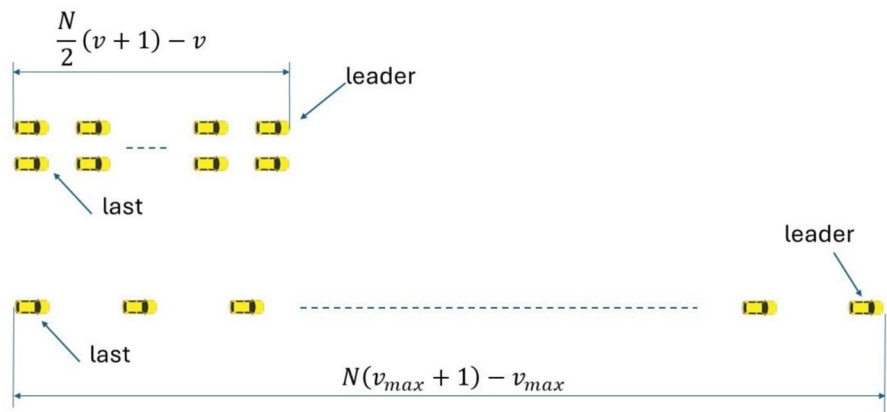
At the end of the period, the distance between the leader and last car is  $\frac{N}{2}(v + 1) - v - z + y$ . We compute  $x$  such that this distance is large enough to let the last car accelerates to  $v_{max}$  (hence stabilizing the length of the resulting platoon). Because all the cars in the resulting platoon are at speed  $v_{max}$  the length of the platoon must be at least  $N(v_{max} + 1) - v_{max}$  see Fig. 6, hence

$$\frac{N}{2}(v + 1) - v - z + y = N(v_{max} + 1) - v_{max}.$$

Solving for  $x$  leads to

<sup>1</sup> In the sense that the traffic pattern follows the long-term dynamic  $d = v_{max}$  as defined and shown to be a recurrent pattern in Cohen et al. (2022, 2023)

**Fig. 6** Merging two lines needs to make room, illustration. At the top, the initial configuration, two parallel lines of  $N/2$  cars at speed  $v$  and head-tail distance  $d = v$ . At the bottom, the optimal final configuration, the  $N$  cars in a single line at maximal speed  $v_{max}$  and head-tail distance  $d = v_{max}$



$$x = \frac{N}{2} \left( 1 + \frac{v_{max} + 1}{v_{max} - v} \right) - 1$$

Therefore the number of time steps for the last car to exit the zone C (after reaching the maximal speed) is bounded by  $x + (v_{max} - v - 1)$ , leading to

$$\text{Number of time steps} \geq \frac{N}{2} \left( 1 + \frac{v_{max} + 1}{v_{max} - v} \right) + v_{max} - v - 2 \tag{3}$$

This bound does not consider the time needed for managing the cars between the first and the last and assume the reached traffic pattern is optimal (the head tail distance  $d$  is constant equal to  $v_{max}$ ), see Fig. 6. We observe that the multiplying factor of  $N$  grows as  $v$  increases. This again leads to the conclusion that a small entry speed is favorable. Here, it reduces the time spent in zone C. Interestingly, our detailed analysis leads to the bound (8) that shows the same asymptotic behavior as  $N$  grows.

### 4 Informal presentation of the merging algorithm

In the previous section, we have considered only the merging of  $N/2$  cars on-ramp with  $N/2$  cars on the main road. We illustrate the merging algorithm using Fig. 7. In this figure are represented some speed profiles at times  $t = 0, N_t, P$  (defined below). On the  $x$ -axis are the road positions and on the  $y$ -axis the speed of the cars on the mainroad at this position. The point  $M$  is the merging point, at which regulator's car switch from car following to merging.

We denote time  $t = 0$  the time at which the merging algorithm starts. Cars arriving at the merging point  $M$  reduce their speed from  $v_{max}$  to  $v$ ,  $v$  being the speed of the cars on the on-ramp. This speed profile will propagate upstream<sup>2</sup>. This means that the trailing cars all decrease

<sup>2</sup> This is similar than a *wide moving jam* in the three-phase traffic theory (Kerner 2021).

their speed to  $v$  and the position at which this happen goes backward on the road. This is illustrated by the arrow pointing backward ( $\Leftarrow$ ) on Fig. 7. This can be made precise by leveraging the description of the car-following model given by (1). We introduce  $x^n(t)$  the position of the  $n$ -th car at time  $t$ . It is known that  $x^n(t) - x^{n+1}(t) - 1 = v^n(t) + k$ , where<sup>3</sup>  $k \geq 0$  is a constant of the motion (it is our assumption that the flow is synchronized), see (Cohen et al. 2022, 2023). Hence, if at time  $t$  the  $n$ -th car decelerates  $v^n(t) = v^n(t - 1) - 1$  at position  $x^n(t)$ , the  $(n + 1)$ -th car decelerates at time  $t + 1$ ,  $v^{n+1}(t + 1) = v^n(t)$  at the position  $x^{n+1}(t + 1) = x^{n+1}(t) + v^n(t) = x^n(t) - 1 - k < x^n(t)$ .

Notice that the speed of the propagation of the wave depends on the distances between the cars (the constants  $k$ ). In the following it is assumed that  $k = 0$  corresponding to a maximum flow and a wave propagation speed of 1.

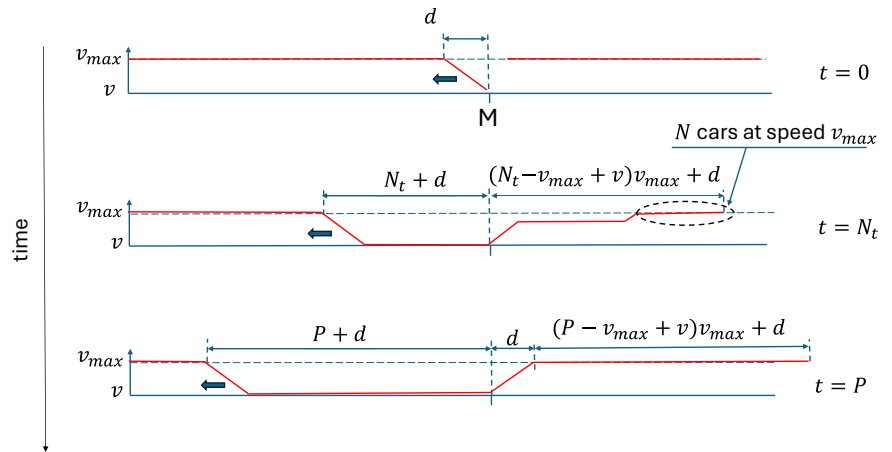
Then, after the merging point, cars accelerate and the two lines are merged. To *make room*, the first car accelerates back to  $v_{max}$  and the trailing cars accelerate first to  $v_{max} - 1$  to make room<sup>4</sup> and finally to  $v_{max}$  when the distance with the leading car is large enough. This process stops at time  $t = N_t$  when the  $N$  cars executing the merging algorithm are at speed  $v_{max}$ , see the middle of Fig. 7. In the following we call this period of time from  $t = 0$  to  $t = N_t$  the *Phase I*, see also Fig. 15. Notice that at time  $N_t$  the cars upstream to the  $N$  cars executing the merging algorithm are at speed less than  $v_{max}$ . At time  $N_t$  the last of the  $N$  cars accelerate to  $v_{max}$  and the cars upstream starts to accelerate later depending on the head-tail distances. This is crucial. Indeed, the merging algorithm can be restarted only when all the cars upstream are at maximal speed  $v_{max}$ .

At the end of phase I,  $N/2$  cars have merged with the mainstream flow. The algorithm should be repeated to merge successive batch of cars on-ramp. This repetition

<sup>3</sup> The distance between two cars is the number of free slots between them.

<sup>4</sup> Notice that this speed profile is a bit different than the one we consider to compute the lower bound (3).

**Fig. 7** Some speed profiles at times  $t = 0, n_t, P$ . On the x-axis is the road positions, on the y-axis the speed of the cars on the mainroad. The distance  $d$  is the distance between the last car at speed  $v_{max}$  and the leader at speed  $v$ ,  $d = (v_{max} + v + 1)(v_{max} - v)$



requires that cars on the zone C after the merging point  $M$ , see Fig. 3, are all at maximal speed since our merging algorithm requires that the leading car accelerates to  $v_{max}$ . At time  $t = N_t$  we observe on the middle of Fig. 7 that the cars following the  $N$  merged cars are at speed  $v_{max} - 1$ . The following cars are cars coming from the mainstream flow. With our merging algorithm it takes  $P - N_t$  time units before all the cars after the merging point  $M$  are at maximal speed  $v_{max}$ . This period of time between  $t = N_t$  and  $t = P$  is covered by what we call Phase II and Phase III in the rest of the paper, see also Fig. 15. The duration of phases II and III are estimated in Sections 6.1 and 6.2. The estimate for  $P$  is given by (14) and for  $N_t$  by (8).

In summary at time  $t = P$  the merging algorithm can restart and a new batch of  $N/2$  cars from on-ramp merged with the mainstream flow.

From the global point of view, we observe that there is a traffic pattern that propagates upstream, the traffic pattern on the bottom on Fig. 7 of length  $P + 2d$  where cars move at speed  $v$ . This traffic pattern will interact with traffic incoming from an on-ramp upstream, preventing cars to apply the merging algorithm (since cars maybe forced to stay at speed  $v$  instead of accelerating to  $v_{max}$ ). This shows that the access from different on-ramps to the mainstream lane should be coordinated. This is illustrated by Fig. 8

In terms of platoons, an on-ramp maybe seen as dividing a stream into platoons. As shown on the top of Figs. 7 and 8 the car that starts the merging algorithm decreases its speed from  $v_{max}$  to  $v$  and then accelerates back to  $v_{max}$ , the car surrounded by red in Fig. 9. The car in front of it continues at speed  $v_{max}$ . In the end an empty region of length  $v_{max} + (v_{max} - v)^2$  occurs between the two platoons, see Fig. 9. Indeed, the speed of the car leading the merging algorithm follows the sequence  $v_{max} - 1, \dots, v, v + 1, \dots, v_{max}$  while the car in front of it continues at speed  $v_{max}$ . The speed differences increase the distance between the two cars by  $(v_{max} - v)^2$  (initially the distance is assumed to be  $v_{max}$ ).

On the newly created platoons a speed profile of the form of the one depicted on the bottom of Fig. 7 propagates upstream.

As we emphasize in this Section, the description of the creation and gathering of platoons as well as the propagation of speed profiles is mandatory to a deep understanding of the global traffic patterns. Now we focus on the detailed description of a merging algorithm and the analysis of the dynamic of more global mechanisms is postponed to subsequent works.

### 5 Algorithm

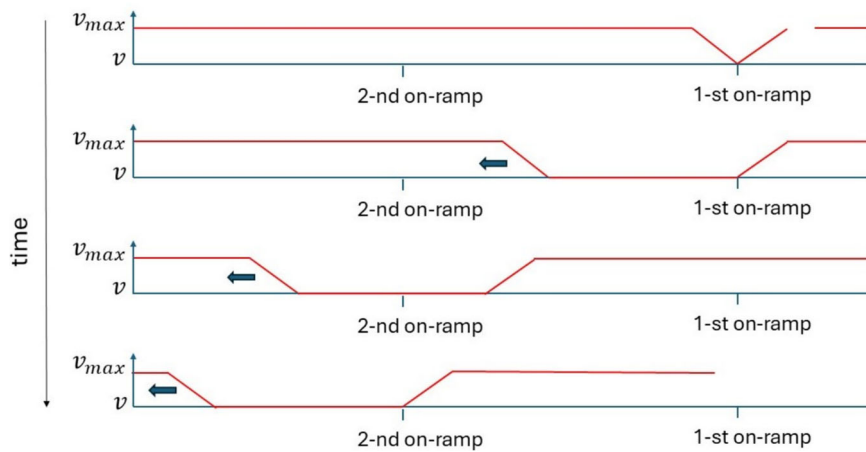
We consider a section of the road composed of the target lane  $n_1$  and the second lane  $n_2$  of size  $L_2 \in \mathbb{N}$ , see Fig. 4 and (4). Vehicles on lane  $n_2$  must move onto lane  $n_1$ . In this section we do not consider upstream cars on  $n_1$ , we restrict our attention to the  $N$  cars involved in the merging algorithm. We represent a configuration at a time  $t$  on the road in the following way:

$$\begin{aligned} n_1 & \{ \dots 3 \dots 5 \dots 5 \dots \} \\ n_2 & \{ \dots 2 \dots 4 \dots \} \end{aligned} \tag{4}$$

where a point  $.$  denotes an empty cell and a number denotes the speed of a vehicle present in that cell.

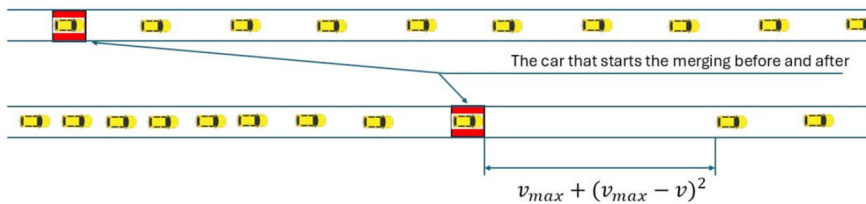
#### 5.1 Initial conditions

The description of the algorithm is limited to the cars involved in the merging algorithm. The cars upstream are assumed to be driven by the car-following speed regulator, i.e. they maintain the head-tail distance  $d = v$  with the leading car, and are discussed later in Section 6. The speed  $v$  is imposed by the algorithm but its value is not yet specified. It is when a car enters one of the section  $n_1$  or  $n_2$  of the road that the regulator switches to the merging mode and applies the algorithm we describe in this section.



**Fig. 8** The propagation of the wave of decelerating cars upstream prevents the merging algorithm to be started on a 2nd on-ramp located upstream. To restart the merging algorithm it is necessary to let the wave pass. **From top to Bottom:** The leading car starts the merging

algorithm by changing its speed following the sequence  $v_{max} \rightarrow v \rightarrow v_{max}$ . Next, the merging is finished. Next, The wave propagates and prevents the merging to be applied at the 2nd on-ramp. Last, The merging can start at the second on-ramp



**Fig. 9** Top: A continuous stream of cars composing a platoon cars move at speed  $v_{max}$ . Bottom: After one car started and executed the merging process, thus decelerating to speed  $v$  and then accelerating to

$v_{max}$ . On the figure the leader car is back at speed  $v_{max}$  and the following cars are still accelerating. We observe two platoons

We assume that cars are coordinated to an initial configuration<sup>5</sup> in which the vehicles are at equal speed and distance. For example, with  $N = 8$  then we can have the following configurations:

$$A \begin{cases} 1.1.1.1..... \\ 1.1.1.1..... \end{cases} \quad B \begin{cases} 2..2..2..2..... \\ 2..2..2..2.. \end{cases} \quad (5)$$

Given the initial condition, a merging sequence is defined (Fig. 10). For example, for configuration **A** the exit order is:

leading to a final configuration

..8...7...6...5...4...3...2...1.....

where all cars move at speed  $v_{max}$  with head-tail distance  $d = v_{max}$ . We denote by  $q_a$  the sequence number of car  $a$ .

### 5.2 Description of the algorithm

*Algorithm 1* works as follows. First, time  $t$  is set to 0 at the initial configuration (a traffic pattern like (5)). Let  $a$  be a

vehicle and  $q_a$  its index in the merging sequence. If  $t$  is less than  $q_a$ , then  $a$  maintains its current speed  $v_a = v$ . If  $t$  is equal to  $q_a$ , then  $v_a$  increases by one unit. If, in addition, its position is even, then  $a$  changes lanes. Note that the vehicles on lane  $n_2$  always have an even index in the merging sequence. Finally, if  $t$  is greater than  $q_a$ , then there are three different cases.

- If  $v_a$  is less than  $v_{max} - 1$  then  $v_a$  increases by one unit.
- If  $v_a$  equals  $v_{max} - 1$  and the head-tail distance with the leading car is smaller ( $\leq$ ) than  $v_a$ , then  $v_a$  stays constant.
- If  $v_a$  equals  $v_{max} - 1$  and the head-tail distance with the leading car equals  $v_a + 1$ , then  $v_a$  increases by one unit.

It is useful to note that the number of steps  $a$  stays at speed  $v_{max} - 1$  is seen in Appendix Proposition 1 to be

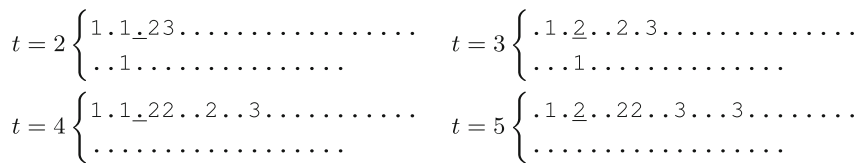
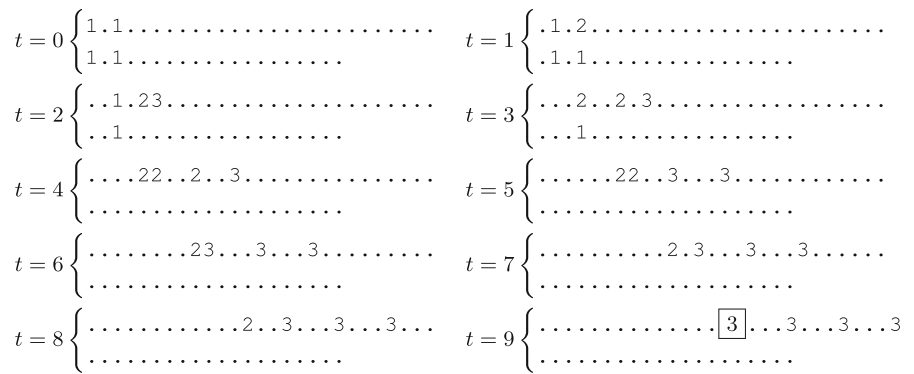
$$\lfloor \frac{q_a}{2} \rfloor (v + 1) + 1 \quad (7)$$

with  $q_a$  the position in the merging sequence.

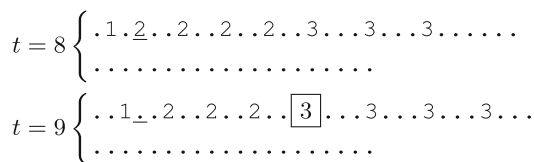
<sup>5</sup> Using a strategy similar to one described in Section 2.



**Fig. 11** Example of execution of the merging algorithm with  $v_{max} = 3$ ,  $v = 1$  and  $N = 4$ . At time 9 the algorithm stops when all cars involved in the merging algorithm gets to maximal speed. The last one is boxed at time 9. For further reference we call it *last*. Notice that *last* stays at speed 2 ( $=v_{max} - 1$ ) 5 time units as predicted by (7), i.e.  $q_4 = 5$



**Fig. 12** Example of Fig. 11 completed to include the incoming flow in the merging area. The merging point is underlined. Algorithm 1 can restart at the time when cars arriving at this point can accelerate to  $v_{max}$  without restricted by cars down stream at lower speed



**Fig. 13** The cars upstream to *last* (boxed in the figure) cannot accelerate to  $v_{max}$  as they enter the merging area. Hence, Algorithm 1 cannot be restarted

The example in Fig. 11 has to be completed to take into account the upstream flow of cars entering continuously the merging area, see Fig. 12.

When comparing Fig. 12 to Fig. 11, we observe at time 2 and 5 the appearance of upstream cars in the mainstream. This stream is composed of cars at speed 1 (in general  $v$ ) and head-tail distance 1 ( $d = v$ ). What is important to notice at time 5 is that the car on the merging point cannot accelerate because of the cars downstream that apply the merging algorithm and are stuck at speed 2 (in general  $v_{max} - 1$ ). *This prevents the merging algorithm to start again.* Indeed, the first car of the merging sequence accelerates regularly to speed  $v_{max}$  (which is not possible here), see Fig. 13.

In Fig. 13 we see that when the merging algorithm finishes at time 9, the last car that participates to the merging algorithm (represented in a box) is at maximal speed 3, but it is followed by a line of cars at lower speed (in general  $v_{max} - 1$ ). *To restart the merging algorithm requires to wait for all these cars get to speed  $v_{max}$ .*

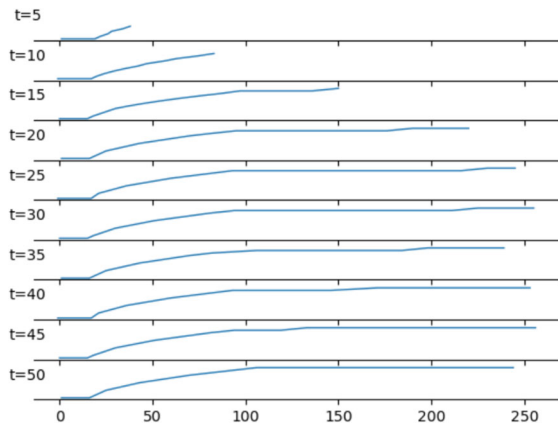
Before turning to the analysis of the performance of Algorithm 1 we describe the traffic dynamics as plotted in

Fig. 15. In this figure we see the states of the cars on road as the merging algorithm is run until the time it can be run again. There,  $N = 6$  car participating to the merging process. We plot in magenta the last such car and for further reference we denote it by *last*. The cars plotted on red are the ones not yet in the merging area. Such cars move at speed  $v$  and there is an infinite number of them coming upstream.

In the figure there are 3 phases to be distinguished. Phase I correspond to the execution of Algorithm 1 as described in Section 5.2. We see that at the end of this phase the 6 cars involved in the merging process are at maximal speed, the cars are represented by squares instead of dots when their speed is maximal ( $v_{max}$ ).

We observe that at the end of phase I there are cars behind *last* that are in the merging area but cannot accelerate to  $v_{max}$ . Repeating Algorithm 1 requires to wait that there is no constraint on the acceleration for these cars. These are Phase II and III. Actually, after *last* accelerates the cars behind start to accelerate one after the other and Algorithm 1 can be repeated when the car entering the merging area can accelerate straightaway.

Analysing Phases II and III requires to understand the speed at which the wave of accelerating cars move. It happens that in Phase II the speed is 1 (cell per unit time) - this is plotted as  $slope_1$  - and in Phase II it is  $(v_{max} - 1)/v$  (cells per unit time)—this is plotted as  $slope_2$ , see Equation 13. This is due to the fact that in Phase II the cars move while preserving the invariant that the head-tail distance equals the speed. Hence, if a car accelerates at position  $x$  the following car accelerates at position  $x - 1$ , it



**Fig. 14** The x-axis is the road positions and on the y-axis the car speeds. We observe that until time  $t = 50$  there are cars stuck at speed  $v_{max} - 1$  waiting for extra inter-distance. This prevent the restart of the merging algorithm

results a wave of acceleration that moves the road backward at speed one. In Phase III the pattern is different as illustrated in Fig. 16.

The time to complete phases I, II and II are respectively given by Equations (8), (11), (13).

This is also illustrated in Fig. 14. We observe that some cars stay at speed  $v_{max} - 1$  for some times, waiting for the head-tail distance to get large enough (Fig. 15). This prevents the merging algorithm to be restarted since this would require that the leading car accelerates up to speed  $v_{max}$ .

### 6.1 Duration of phase II

It is crucial to understand the traffic patterns of the line of cars behind the last car in the merging sequence. We denote *last* for further reference this car.

First, at the time when *last* accelerates it is at distance  $Nv - \frac{N}{2}(v + 1) + v = N\frac{v-1}{2} + v$ ,

from the merging point i.e., it starts at distance  $\frac{N}{2}(v + 1) - v$  with a sequence of  $N$  steps at speed  $v$ . At this time there are

$$k = \frac{v-1}{v+1} \left( \frac{N}{2} + 1 \right)$$

cars between *last* at the merging point which are all at speed  $v$  and distance<sup>6</sup>  $d = v$ .

Because *last* starts accelerate at distance  $N\frac{v-1}{2} + v$  from the merging point it will last  $N\frac{v-1}{2} + v$  steps before cars enter the merging area and are able to accelerate (the wave of acceleration goes back with one distance unit per time unit because  $d = v$ ). During this period the flow of cars entering is  $v/(v + 1)$ , hence it will be

$$k' = k + \left( N\frac{v-1}{2} + v \right) \frac{v}{v+1} \tag{11}$$

cars behind *last* whose head-tail distance will always equal the speed.

### 6.2 Duration of phase III

At the time when cars enter the merging area and accelerate the traffic pattern changes. Actually, when the cars enter at speed  $v$  there are  $v - 1$  cars entering successively (at times  $1, 2, \dots, v - 1$  and the  $v$ -th car enters at time  $v + 1$ ). Hence when the car's speed stabilize at  $v_{max} - 1$  there will be  $v - 1$  head-tail distances  $d = v_{max} - 1$  followed by one with  $d = v + 2 + 2(v_{max} - v - 2)$ , see Fig. 16 for an illustration with  $v_{max} = 5, v = 2$ .

For this traffic pattern the wave of accelerating cars progresses upstream at speed  $(v_{max} - 1)/v$ . Notice this is faster than 1 the speed of the wave when  $d = v$ .

We denote  $d_{end}$  the distance from the merging point to *last* (the one boxed in Fig. 13) when the merging algorithm stops, i.e. the position where the speed of *last* increases from  $v_{max} - 1$  to  $v_{max}$ . The wave of accelerating cars starts at this point up to the merging point. The algorithm can restart when the wave meet the cars accelerating from the merging point (at distance  $(v + v_{max})(v_{max} - v + 1)/2$  from the meeting point). This distance is

$$d'_{end} = d_{end} - (v + v_{max})(v_{max} - v + 1)/2$$

It can be shown, see Proposition 3 in Appendix that

$$d_{end} = N\frac{(v-1)}{2} + v + (v+1) + \dots + (v_{max}-2) + \left( \frac{N}{2}(v+1) + 1 \right) (v_{max}-1) \tag{12}$$

$$d'_{end} = \frac{N}{2} (v-1 + (v+1)(v_{max}-1)) + v.$$

and the time to reach this configuration is given by (8).

The wave of accelerating car starts from  $d_{end}$  and goes back  $k'$  unit distance in the first  $k'$  time unit because the speed of the wave is 1 since  $d = v$  between the  $k'$  cars. The remaining distance is  $d'_{end} - k'$  that the wave crosses at speed  $(v_{max} - 1)/v$  for a time

$$\frac{v}{v_{max}-1} (d'_{end} - k'). \tag{13}$$

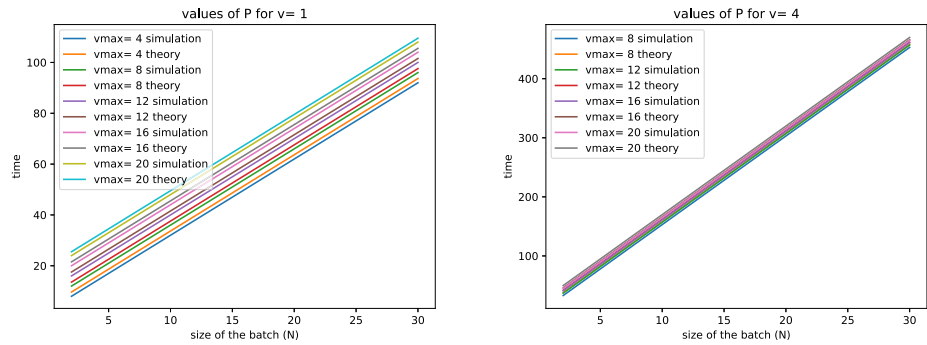
Hence, the algorithm can restart after

$$P = N_t + k' + \frac{v}{v_{max}-1} (d'_{end} - k'). \tag{14}$$

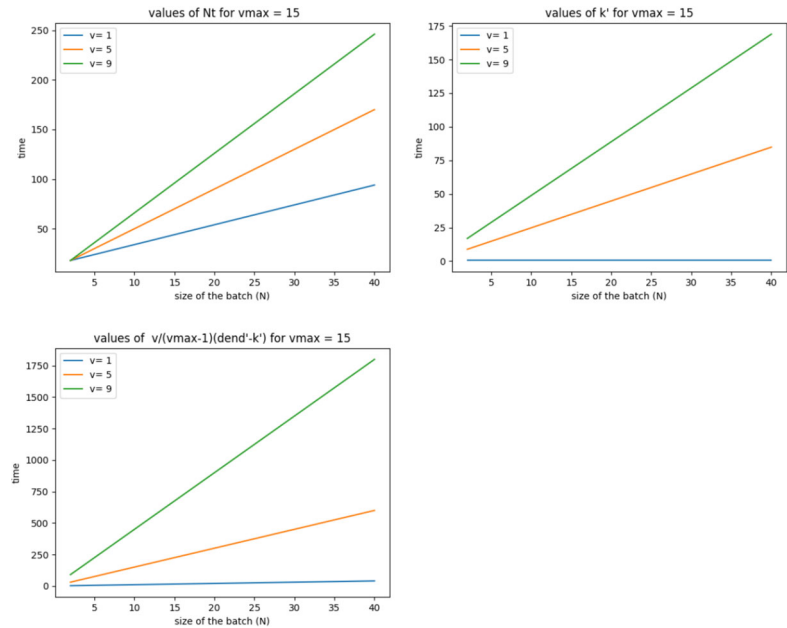
<sup>6</sup> Solve  $k(v + 1) = N\frac{v-1}{2} + v - 1$ , see (10).



**Fig. 17** Numerical validation of (14), the results of simulations are compared to the bound (14)



**Fig. 18** Plot of the 3 summands entering (14), for  $v_{max} = 15$ . Top Left: Execution time of Algorithm 1  $N_t$ , see (8) corresponds of the duration of Phase I. Top Right: Equation (11). Corresponds to the duration of Phase II Bottom: Equation (13) corresponds to the duration of Phase III. Phase I, II, and III are illustrated in Fig. 15



$$\frac{N}{2P} = \frac{1}{2 + 3v + v^2}.$$

For  $v = 1, 3, 5$  we get  $\frac{N}{2P} \approx 0.16, 0.023, 0.009$  respectively, that are values close to what can be measured in Fig. 19.

### 8 Conclusion

The analysis conducted in this paper shows that the performance of merging algorithms in a global setting involves many aspects of the traffic pattern dynamics that are not easy to grasp all together. But, from the point of view of optimizing the process it is mandatory to understand the global process. If our analysis does not lead to a comprehensive rule of thumb we believe it is relevant in showing a way to analyze the problem. Moreover, our approach allows us to consider the merging of batches of cars and the conditions to repeat the process.

Our analysis leads to relevant conclusions. In particular, it shows that it is important to control the speed of the cars arriving

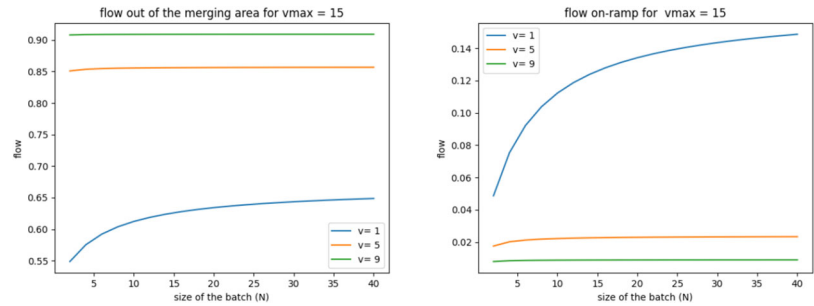
to the merging area. This speed  $v$  should be as low as possible. Notice that Equation (9) leads to the same conclusion since the length of the double road infrastructure increases with  $v$ .

From a technical point of view, the derivation of the estimate (14) shows that it is possible to express the speed of the front wave of acceleration/deceleration of traffic patterns. These speeds are functions of the head-tail distances between the cars. This is an improvement with respect to Cohen et al. (2022, 2023).

As for limitation, our model should consider the more realistic situations where cars occupy more than one cell. This is paramount since it makes possible to consider realistic acceleration. Moreover, it is important to add this new parameter and see whether it reinforces or not our conclusions.

We mention that we do not assert that Algorithm 1 is optimal. Actually, before doing its complete analysis it was not clear in what respect this algorithm should be optimized. This paper suggests that what should be optimized is the period  $P$  given by (14).

**Fig. 19** Left: The total flow after the merging area  $\frac{v}{v+1} + \frac{N}{2P}$ . Right: The flow on-ramp  $\frac{N}{2P}$



From an heuristic point of view, to optimize Algorithm 1 we may exploit the fact that more than one car at a time can increase its distance. This may lead to the factor  $1/(v_{max} - v)$  in Equation (3) that multiplies  $N$  and is missing in (8). More generally, formalizing the problem as an optimization problem for solution would be as well relevant. We will consider such directions for further works.

Finally, we qualified this work as global to stress that we quantify the impact of the lane merging algorithm on input and output flows. However, we are still some steps back from the global picture as shown in Fig. 3. We do not discuss the impact of the slowing area  $B$  before the merging lane that leads to a wave of cars slowing down upstream. In particular in our context of saturated traffic it is likely that the area  $A$  of cars at cruise speed cannot settle and the area  $B$  spread upstream. It is relevant as well to see how the traffic from many successive on-ramps can be synchronized. We keep this for further works.

## Appendix

### Proofs of (7), (8), (12).

**Proposition 1** When car  $a$  applies Algorithm 1 it stays at speed  $v_{max} - 1$  for

$$\lfloor \frac{q_a}{2} \rfloor (v + 1) + 1,$$

where  $q_a$  is the number of  $a$  in the merging sequence, see (3).

**Proof** The proposition is true for  $q_a = 1$  because the first car accelerates from  $v$  to  $v_{max}$  one speed unit per time unit. Let us consider car with  $q_a = 2$ . At the time  $a$  accelerates it is at speed  $v + 1$  and distance 0 from the leading car which is at speed  $v + 2$ , see the example (11). It takes  $v_{max} - v - 2$  time steps for the leading car to reach speed  $v_{max}$  (observe that  $v_{max} \geq v$ ). At this time  $a$  is at distance  $v_{max} - v - 2$  from the leading car and at speed  $v_{max} - 1$ . It stays  $v + 1$  more time units at speed  $v_{max} - 1$  before the distance is  $v_{max}$  and  $a$  accelerates to  $v_{max}$ . In total,  $a$  was  $v + 1 + 1 = \lfloor \frac{2}{2} \rfloor (v + 1) + 1$  time units at speed  $v_a$ . Let us consider the next car  $q_a = 3$  and we will see how the

formula generalizes.  $a$  accelerates one time units after the leading car and always ensures that  $v_a = d$ , i.e. the speed of  $a$ ,  $v_a$ , equals the distance with the leading car. Hence it stays at speed  $v_{max} - 1$  the same number as the leading car, which is what the formula says. This is true for all cars with  $q_a$  odd. Finally, for  $q_a = 4$  the same argument than the one for  $q_a = 2$  applies and the car stays at  $(v_{max} - 1), (v + 1)$  time units more than the leading car. This argument holds for all cars with  $q_a$  even.  $\square$

**Proposition 2** The merging Algorithm 1 takes

$$N_t = N + v_{max} - v + \frac{N}{2} (v + 1) = N \left( 1 + \frac{v + 1}{2} \right) + v_{max} - v$$

before the last car reach speed  $v_{max}$ .

**Proof** The last car has merging sequence number  $N$ . It stays at speed  $v$  for  $N$  time steps, then increases its speed to  $v_{max}$  staying  $\frac{N}{2} (v + 1) + 1$  at speed  $v_{max} - 1$ . In total we get the given expression for  $N_t$ .  $\square$

**Proposition 3** The expression for  $d_{end}$  the distance of the last car involved in the merging sequence at the time its speed reaches  $v_{max}$  from the merging point is given by

$$d_{end} = N \frac{(v - 1)}{2} + v + (v + 1) + \dots + (v_{max} - 2) + \left( \frac{N}{2} (v + 1) + 1 \right) (v_{max} - 1).$$

**Proof** In Proposition 2 we count the number of time steps the last car is involved in the merging process. Similarly, we can count the distance spanned. It is  $Nv$  for the  $N$  first steps followed by the acceleration and stays at speed  $v_{max} - 1$ . This leads to

$$Nv + (v + 1) + \dots + (v_{max} - 2) + \underbrace{v_{max} - 1 + \dots + v_{max} - 1}_{\frac{N}{2}(v+1)+1} \tag{15}$$

We conclude by noting that the last car starts a distance  $\frac{N}{2} (v + 1) - v$  ahead from the merging point, hence this has to be subtracted from (15).  $\square$

### Speed of waves of accelerating cars

We consider the traffic pattern where an infinite line of cars move at speed  $v$  with head-tail distance  $d = v$  between the cars like in following Figure where  $v = 2$ .

.2..2..2..2..2..2.....

If the leading car accelerates from speed  $v$  to  $v_{max}$ , here  $v_{max} = 4$  this leads to the following configurations.

.2..2..2..2..2..3.....

2..2..2..2..2..3..4.....

..2..2..2..2..3..4...4.....

.2..2..2..2..2..4...4...4.....

We observe that the positions at which the cars reaches speed  $v_{max} = 4$  goes upstream 1 cell per time unit. This is general and due to the fact that the head-tail distance  $d = v$  and the distance counts the number of unoccupied cells between the cars. The speed of the wave of accelerating cars is then 1 cell per time unit.

In the case where the distance  $d = v + k$  we observe that we have the configuration (with  $v = 2, k = 1, v_{max} = 4$ )

.2...2...2...2...2...2.....

Which leads to

.2...2...2...2...2...3.....

2...2...2...2...2...3..4.....

..2...2...2...2...3..4...4.....

2...2...2...2...2...4...4...4.....

We observe that the condition  $d = v + k$  is invariant. Hence, at the time the leading car reach speed  $v_{max}$ , the trailing car follows at speed  $v_{max} - 1$  and distance  $v_{max} - 1 + k$ . It follows that after the position update the trailing car reach a position which is  $v_{max} - 1 + k - (v_{max} - 1) = k$  cells behind the position the trailing car was when it reached speed  $v_{max}$ . Hence, the speed of the accelerating wave is  $k + 1$ .

In the situation described by (16) there are  $v - 1$  successive head-tail distances equal to  $d = v$  and 1 at distance  $v + 2(v_{max} - v - 2) + 2 = v_{max} - 1 + k$  with  $k = v_{max} - v - 1$ . This last one leads to an acceleration of  $k + 1 = v_{max} - v$ . Hence in average the speed is  $(v_{max} - 1)/v$  cells per unit time.

To end, we show that in the situation described by (16) the long distance is given by  $v + 2(v_{max} - v - 2) + 2$ . The long distance occurs because one trailing car does not accelerate straight after the leading car but one time unit later. The leading car speeds are of the form  $(v + 1) (v +$

$2) \dots (v_{max} - 1) (v_{max} - 1) (v_{max} - 1)$  for  $(v_{max} - v + 1)$  time units. The trailing car speed are of the form  $v (v + 1) \dots (v_{max} - 1)$ . Hence, between the two cars the distance increases by  $2(v_{max} - v - 2) + 2$  and the total distance is then  $v + 2(v_{max} - v - 2) + 2$ .

**Funding** Open access funding provided by University of Geneva.

**Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

### References

Bayen A, Monache MLD, Garavello M, Goatin P, Piccoli B (2022) Control problems for conservation laws with traffic applications: modeling, analysis, and numerical methods. Springer Nature

Cao W, Mukai M, Kawabe T, Nishira H, Fujiki N (2015) Cooperative vehicle path generation during merging using model predictive control with real-time optimization. *Control Eng Pract* 34:98–105

Charlottin T, Varotto S, Buisson C (2024) Detecting car-following and platoons within traffic using the dynamic time warping algorithm. In: Heart 2024, 12th symposium of the European association for research in transportation

Chen J, Lin L, Jiang R (2017) Assigning on-ramp flows to maximize capacity of highway with two on-ramps and one off-ramp in between. *Phys A* 465:347–357

Cohen N, Chopard B, Leone P (2022) Maximum traffic flow patterns in interacting autonomous vehicles. In: International conference on cellular automata for research and industry, pp 281–291. Springer (2022)

Cohen P, Chopard B, Leone P (2023) Optimal safe driving dynamics for autonomous interacting vehicles. *Nat Comput*, pp 1–13

Daganzo CF (2006) In traffic flow, cellular automata= kinematic waves. *Transp Res Part B Methodol* 40(5):396–403

Guzmán H, Lárraga M, Alvarez-Icaza L, Huerta F (2015) On-ramp traffic merging modeling based on cellular automata. In: 2015 IEEE European modelling symposium (EMS), pp 103–109. IEEE

Huang Z, Zhuang W, Yin G, Xu L, Luo K (2019) Cooperative merging for multiple connected and automated vehicles at highway on-ramps via virtual platoon formation. In: 2019 Chinese control conference (CCC), pp 6709–6714

Kerner BS (2021) Understanding real traffic. Springer

LeVeque RJ, Leveque RJ (1992) Numerical methods for conservation laws, volume 214. Springer

Lighthill MJ, Whitham GB (1955) On kinematic waves i. flood movement in long rivers. *Proc R Soc Lond Ser A Math Phys Sci* 229(1178):281–316

- Linghui X, Jia L, Ran B, Yang F, Zhang J (2019) Cooperative merging strategy for connected vehicles at highway on-ramps. *J Transp Eng Part A Syst* 145(6):04019022
- Lin L, Jiang R, Hu M-B, Wu Q-S (2012) Optimizing traffic flow on highway with three consecutive on-ramps. In: 2012 fifth international joint conference on computational sciences and optimization, pp 514–518. IEEE
- Milanes V, Godoy J, Villagra J, Perez J (2011) Automated on-ramp merging system for congested traffic situations. *IEEE Trans Intell Transp Syst* 12(2):500–508
- Newell GF (1993) A simplified theory of kinematic waves in highway traffic, part i: General theory. *Transport Res Part B Methodol* 27(4):281–287
- Newell GF (2002) A simplified car-following theory: a lower order model. *Transp Res Part B Methodol* 36(3):195–205
- Ntousakis IA, Nikolos IK, Papageorgiou M (2016) Optimal vehicle trajectory planning in the context of cooperative merging on highways. *Transp Res Part C Emerg Technol* 71:464–488
- Petig T, Schiller EM, Suomela J (2018) Changing lanes on a highway. In: 18th workshop on algorithmic approaches for transportation modelling, optimization, and systems (ATMOS 2018). Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik
- Rios-Torres J, Malikopoulos AA (2016) Automated and cooperative vehicle merging at highway on-ramps. *IEEE Trans Intell Transp Syst* 18(4):780–789
- Sicheng Sun X, An JZ, Li P, Shao H (2021) Modeling and simulation of lane-changing management strategies at on-ramp and off-ramp pair areas based on cellular automaton. *IEEE Access* 9:35034–35044
- Zhu J, Easa S, Gao K (2022) Merging control strategies of connected and autonomous vehicles at freeway on-ramps: a comprehensive review. *J intell Connect Veh* 5(2):99–111

**Publisher's Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.