Transportation Research Part E 166 (2022) 102886

Contents lists available at ScienceDirect

Transportation Research Part E

journal homepage: www.elsevier.com/locate/tre

Trajectory planning for autonomous modular vehicle docking and autonomous vehicle platooning operations

Qianwen Li, Xiaopeng Li *

Department of Civil and Environmental Engineering, University of South Florida, Tampa, FL, 33620, United States

ARTICLE INFO

Keywords:
Autonomous modular vehicles
Roadway logistics
Platooning
Docking
Trajectory planning
Pontryagin's maximum principle

ABSTRACT

Emerging autonomous modular vehicle (AMV) technology allows vehicle units to physically dock on or split from each other en route to form vehicles of different lengths. This technology has great potential in roadway logistics where platoons/long trains are formed to transport goods and passengers, i.e., freight and transit systems. AMV docking is an extreme case of autonomous vehicle (AV) platooning in that AMVs are physically connected with zero gaps. This paper formulates the AMV docking and AV platooning trajectory planning problem into a two-stage optimization problem. A feasible cone method is proposed to reveal the theoretical properties of solution feasibility and solve the first-stage problem analytically. This method provides the basics for a parsimonious heuristic approach to design trajectories specified as several quadratic segments. A heuristic alternative solution based on Pontryagin's maximum principle is proposed to solve a special case of the original problem to the exact optimum. Then an exact solution approach based on quadratic programming is proposed to optimize the trajectories. The feasible cone method is used to construct valid cuts to expedite the exact solution efficiency. Numerical experiments show that the parsimonious heuristic approach can achieve near-optimal solutions and greatly reduce the solution time compared with the exact solution approach, appealing to real-time engineering applications. The results also demonstrate the superiority of the parsimonious heuristic approach in optimizing AMV docking and AV platooning trajectories compared with traditional platooning methods. Sensitivity analysis results shed insights into advising parameter selections of platoon-related logistics to balance the tradeoff between operational efficiency and cost.

1. Introduction

For decades, traffic growth outpaced investment in road infrastructure. Traffic congestion ensues and substantially compromises society's economic well-being and long-term sustainability. Emerging autonomous modular vehicle (AMV) technology provides promising mitigation to congestion. The AMV technology allows multiple vehicles to flexibly dock on and split from each other en route with advanced control and communication technologies. This technology is expected to improve the service rate of roadway logistics where long trains/platoons are formed, such as transit and freight transport (Chen et al., 2019).

AMV docking is an extreme case of autonomous vehicle (AV) platooning (Chen et al., 2021; Feng et al., 2019; Li et al., 2021). This technology eliminates car-following gaps via physical connections between multiple vehicles and thus maximizes space efficiency. It

* Corresponding author.
E-mail addresses: qianwenli@usf.edu (Q. Li), xiaopengli@usf.edu (X. Li).

https://doi.org/10.1016/j.tre.2022.102886
Received 11 November 2021; Received in revised form 29 July 2022; Accepted 23 August 2022
Available online 30 August 2022

further smooths vehicle motion trajectories via automated control, particularly during transition stages such as docking and split operations. Trajectory smoothing and improved aerodynamic efficiency by eliminating gaps between docked vehicles result in less congestion, superior energy efficiency, and fewer environmental footprints. Inspired by this technology, various experiments have been proposed and tested recently. Pilot experiments on autonomous rail rapid transit with adjustable train lengths, i.e., different numbers of AMV, are going on in Zhuzhou, China (Lambert, 2017). AMVs designed by Next Future Transportation Inc. are being tested in Dubai (Tarek, 2018) and Singapore (Ackerman, 2016).

Despite such great potential, the trajectory planning of AMVs has not been investigated to the best of our knowledge. If connections need to be drawn from the existing literature, physically connected AMVs can be viewed as an extreme adaption of platooned AVs, i.e., reducing vehicle gaps to zero. AV platooning has drawn much attention from researchers in recent years due to its potential to improve the safety, stability, and efficiency of traffic systems (Li and Wang, 2007; Liu et al., 2021). Research on AV platooning can be traced back more than 50 years (Levine and Athans, 1966; Melzer and Kuo, 1971). Most existing studies on AV platooning focus on operations of vehicles that are already in a platoon, such as string stability (Ploeg et al., 2013), inter-vehicle communication (Tank and Linnartz, 1997), control strategies (Horowitz and Varaiya, 2000; Stankovic et al., 2000) and safety issues (Alam et al., 2014). In addition to passenger cars (Lee and Tomizuka, 2003; Ma et al., 2012; Tan et al., 1998), platooning has been well investigated for heavy-duty vehicles (HDV) for their greater potential for energy consumption reduction (Alam, 2014; Bishop et al., 2014; Bonnet and Fritz, 2000; Larsen et al., 2019; Noruzoliaee et al., 2021; W. Zhang et al., 2017). Examples of recent undertakings on HDV platooning include PATH (Lu and Shladover, 2014), CHAUFFEUR (Bonnet and Fritz, 2000), KONVOI (Kunze et al., 2011), and Energy ITS (Tsugawa, 2013).

Research on transitions between separated vehicles and connected platoons is relatively recent. Research has been conducted regarding how vehicles enter or leave a platoon for better safety performance (Baskar et al., 2008; Milanés et al., 2010). Virtual controllers placed at major intersections in a road network were used to coordinate approaching vehicles to form platoons with appealing energy efficiency (Larson et al., 2014). Liang et al. (2013) addressed a single HDV increasing speed to catch other vehicles or platoons. They proposed a coordination algorithm to platoon several vehicles by coordinating neighboring vehicles pairwise (Liang et al., 2015). Cooperative adaptive cruise control (CACC) has been used to manage vehicle platoon (Amoozadeh et al., 2015; Liu et al., 2018). Morales and Nijmeijer (2016) controlled vehicle trajectories to join the existing platoons without much disturbance to the vehicles already in the platoon. Bang and Ahn (2017) adopted swarm intelligence to control platoon formation and evolution with the spring constant and damping coefficient. Wei et al. (2017) adapted Newell's car-following model to approximate AV platooning by varying the reaction time. Heinovski and Dressler (2018) developed both centralized and distributed platooning models from the perspective of vehicles seeking to join platoons. More recently, a few studies investigated the vehicle platoon formation in a decentralized fashion (Zhuang et al., 2020), i.e., solving every single vehicle's trajectory respectively, or in a receding centralized fashion

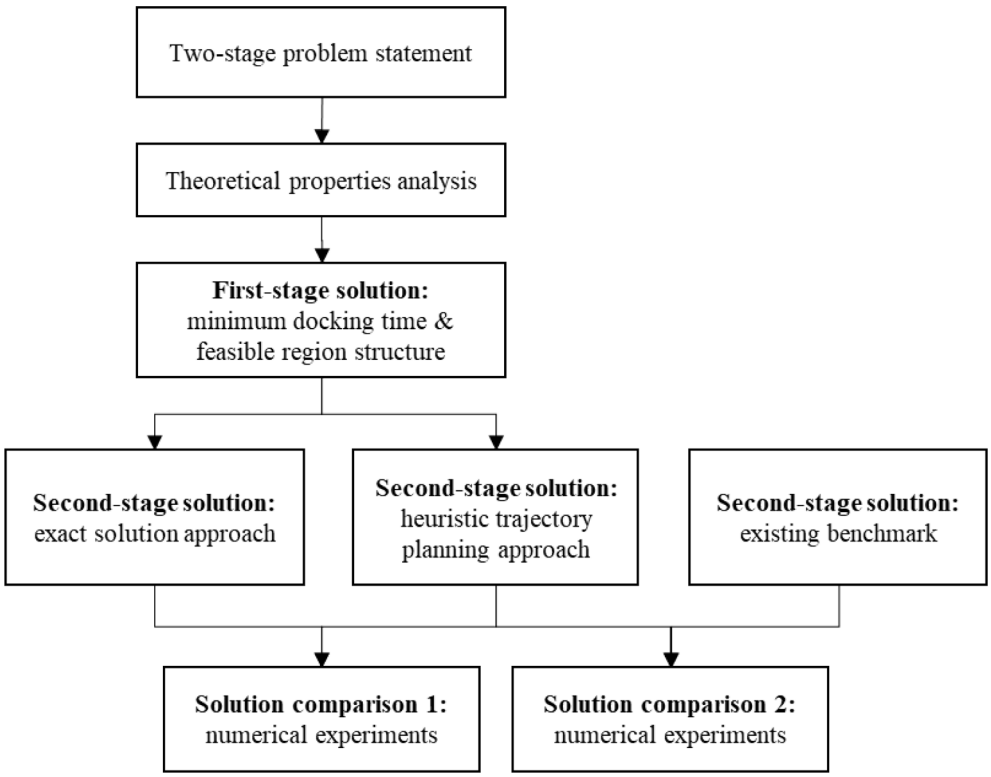


Fig. 1. Methodology framework.

(Firoozi et al., 2021), i.e., solving vehicles' trajectories in a time horizon shorter than the actually needed horizon and repeating this until the platoon is formed. Despite these successes, detailed trajectory planning for multiple vehicles to form platoons simultaneously in a coordinated and optimal manner has been neglected, which limits the efficiency and performance of platooning operations.

Some efforts have been made to optimize trajectories of scattered vehicles to pass roadway segments efficiently in a manner similar to platooning, i.e., the ending speeds and gaps are not strictly the same (Jiang et al., 2017; Jin et al., 2013; Yao et al., 2018). However, little effort has been made to reveal the analytical properties of problem and solution structures, which impedes our understanding of and insights into trajectory planning problems. Li and Li (2019) formulated a general vehicle trajectory optimization problem without regulating the ending speed and car-following distance and analyzed solution feasibility conditions. Yet, without the final platooning conditions, Li and Li (2019) assumed that the operation time horizon was fixed and given rather than a variable to be minimized, as in this paper. When the time horizon is set too long, operational efficiency was not maximized (i.e., the operation could have finished earlier to reduce the potential impedance on traffic and thus enhance mobility), and the platooning benefits were not maximized (i.e., vehicles could have stayed in the platoon for the longest duration after the operation to reduce energy consumption and improve mobility). An arbitrary time horizon could easily lead to infeasibility such that platoons cannot form when the time horizon is too short and thus it must be chosen appropriately, which requires extra effort. Further, Li and Li (2019) did not propose any customized solution approach but only used an off-the-shelf standard solver. A standard solver without customization to the specific problem structure often takes a long time to solve a problem instance, and the solution time is expected to increase significantly with the problem size. A long solution time would disqualify the solution approach from serving real-time vehicle trajectory applications on the time scale of at least a sub-second. Last but not the least, without regulating the final speed and final car-following distance, the solution quality in Li and Li (2019) remains uncertain, especially the final states can be chaotic. To achieve the best objective (i.e., the smoothest trajectories), the final speed in Li and Li (2019) can be rather small, which would seriously impede the mobility of both subject vehicles and surrounding traffic. More importantly, the following vehicle may inevitably collide with the preceding vehicle right after the optimization because the following speed is greater while the distance is short.

Motivated by the above research gaps, this paper proposes an AMV **D**ocking and AV **P**latooning **T**rajectory **P**lanning **P**roblem (DP-TPP) on simultaneous docking and platooning operations of multiple vehicles with general initial conditions (e.g., locations and speeds). We first model the investigated problem into a two-stage nonlinear program. Due to its complex structure, existing methods or solvers cannot solve this model efficiently to satisfy its needs for assisting real-time operations. To tackle this challenge, we propose a methodology framework for devising an efficient heuristic solution approach and verifying its performance with an exact solution counterpart approach, as illustrated in Fig. 1. We first analyze the theoretical properties of solution feasibility by extending the time geography theory. This also yields the optimal solution to the investigated first-stage problem for solving the minimum docking time. The minimum docking time and feasible region structure from the theoretical analysis enable the development of a parsimonious heuristic trajectory planning approach that obtains near-optimal solutions to the DP-TPP as a small number of simple quadratic segments in milliseconds. A heuristic alternative solution based on Pontryagin's maximum principle is also proposed to solve a special case of the DP-TPP to the exact optimum. Further, these theoretical properties enable the formulation of an exact solution approach with quadratic programming and expedite its solution speed with feasibility cuts. Numerical experiments are conducted to show the superior efficiency of the heuristic approach while verifying the near-optimality of the solutions compared to the exact optima. Another set of numerical experiments is conducted to demonstrate the superiority of the heuristic approach compared with traditional platooning methods in optimizing AMV docking and AV platooning trajectories. Finally, sensitivity analyses are conducted to draw managerial insights into platoon-related logistics.

The contributions of this paper are summarized as follows. First, a trajectory planning problem for AMV docking and AV platooning (DP-TPP) is proposed to fill the gap of AMV and AV operations. Second, theoretical properties of the solution feasibility are analyzed to enhance an in-depth understanding of the investigated problem. Third, the minimum docking time and feasible region structure from

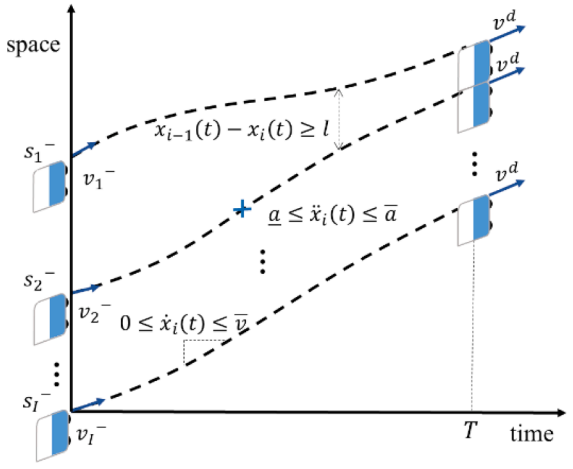


Fig. 2. Problem demonstration.

the theoretical analysis lead to the development of a parsimonious heuristic trajectory planning approach to solve the DP-TPP to near-optima in real-time, appealing to realistic vehicle control. Fourth, A heuristic alternative solution based on Pontryagin's maximum principle is also proposed to solve a special case of the DP-TPP with safety constraints relaxed.

The paper is organized as follows. Section 2 briefly describes the investigated trajectory planning problem. Section 3 analyzes theoretical properties by extending the time geography theory, which leads to the analytical solution to the first-stage problem, i.e., the minimum docking time, and derives the feasible region structure. Section 4 proposes a parsimonious heuristic trajectory planning approach to obtain near-optimal trajectories. Section 5 presents an exact approach to optimizing AMV docking and AV platooning trajectories. Section 6 runs instances to compare the performance of the heuristic approach with the exact solution approach. Numerical experiments prove the superiority of the proposed approach in optimizing vehicle trajectories compared with the existing benchmark, i.e., the CACC. Sensitivity analysis is also conducted to investigate the impacts of key parameters. Last, Section 7 concludes this paper and discusses potential future research directions.

2. Problem statement

This section presents the basic settings of the trajectory planning optimization problem, as illustrated in Fig. 2. The investigated system contains I AMVs/AVs on the same lane, indexed by $i \in \mathscr{I} = \{1, 2, \cdots, I\}$ from downstream to upstream. The length of each vehicle is denoted by l_i. Each vehicle could also be a short platoon that consists of multiple vehicles. The vehicle-to-everything (V2X) communication is assumed to be instantaneous with no lags. These AMVs/AVs send platooning requests to the operation center (e.g., the leading vehicle or a roadside unit). After receiving these requests, the operation center decides whether to approve them based on the current traffic situation (Maiti et al., 2017). For example, if vehicles are within a rather oscillated bottleneck, their platooning requests are unlikely to be approved since the platooning process may dampen the overall oscillation. If platooning requests are approved, the operation center will collect initial vehicle conditions, i.e., speed and location, and plan the corresponding trajectories. How to deal with vehicle operation requests is an interesting research topic yet not the subject of this paper. Readers are referred to other relevant studies (Amoozadeh et al., 2015; Hall and Chin, 2005; Valente et al., 2014). This paper investigates the trajectory planning problem when I platooning requests are approved. The operation center collects AMV/AV i's initial location s_i^- and speed v_i^- at time 0 and plans their platooning trajectories. At time T, all AMVs are docked with bumper-to-bumper connection, or all AVs are platooned with the same platoon gap Δgap and cruise at speed v^{d}. v^{d} and Δgap are application parameters that shall be set by practitioners per application needs, e.g., roadway speed limit and traffic congestion situation. AMV docking is an extreme case of AV platooning with zero platoon gaps. With properly designed docking devices, the physical connections between vehicles can be finished instantaneously and smoothly. Vehicle docking can be easily generalized to AV platooning by adding positive platoon gaps (i.e., equivalent to extending the vehicle effective length). To be specific, if the AMV length is set as $\Delta gap + l_i$ where l_i is the AV length and Δgap is the platoon gap, AMV docking is the same operation as AV platooning. Thus, hereafter, we use AMV-related terminology for expression simplicity. The decision variables are the trajectories for all vehicles at all time points, i.e., $\chi := \{x_i(t)\}_{t\in[0,T],i\in\mathscr{I}}$ where $x_i(t)$ denotes the location of vehicle i at time t. Without loss of generality, the location of a vehicle is measured at its front bumper, and it is set to increase from upstream to downstream. These variables are subject to the following constraints.

1. The speed of each AMV, formulated as the first order derivate of the corresponding location, should be less than the speed limit $\bar{v}$. Also, the speed should be greater than 0 since we assume all AMVs cannot move backward on the road, which yields the following speed constraints:

$$0 \le \dot{x}_i(t) \le \bar{v},\ \forall i \in \mathscr{I}, t \in [0, \infty). \tag{1}$$

2. The acceleration (or deceleration) rate of each AMV, formulated as the second-order derivative of the corresponding location, is bounded as follows:

$$\underline{a} \le \ddot{x}_i(t) \le \bar{a},\ \forall i \in \mathscr{I}, t \in [0, \infty), \tag{2}$$

where $\underline{a}$ denotes the minimum acceleration rate and $\bar{a}$ denotes the maximum acceleration rate. $|\underline{a}|$ is assumed to be equal to $|\bar{a}|$.

3. The safety constraints require that the separation between two consecutive AMVs is always no less than the preceding AMV length (since an AMV's location is measured at its front bumper) during the docking operation:

$$x_{i-1}(t) - x_i(t) \ge l_{i-1},\ \forall i \in \mathscr{I}\backslash\{1\}, t \in [0, \infty). \tag{3}$$

The initial boundary conditions are defined as follows. AMV i starts at its initial location s_i^- and speed v_i^- at time 0.

$$x_i(0) = s_i^-, \dot{x}_i(0) = v_i^-, \forall i \in \mathscr{I}. \tag{4}$$

5. Finally, the following final boundary conditions shall be satisfied. AMV i cruises at a speed v^{d} after being docked (i.e., $x_{i-1}(T)$ and $x_i(T)$ are separated by AMV $i-1$'s length) at final time T:

$$\dot{x}_i(T) = v^{\mathrm{d}},\ \forall i \in \mathscr{I}, x_{i-1}(T) - x_i(T) = l_{i-1}, \forall i \in \mathscr{I}\backslash\{1\}. \tag{5}$$

The investigated trajectory planning problem (i.e., DP-TPP) aims to identify the most appealing trajectory shapes for efficiently implementing the docking operation while improving mobility, riding comfort, and energy efficiency. To capture these factors, we propose the following two-stage model. The first-stage problem (FP) minimizes the docking operation time T for operational efficiency. The reason for selecting this objective is two-fold. First, it is beneficial to mobility because the potential impact of the docking

operation on subject vehicles and other vehicles is minimized. Second, the minimum docking operation time suggests that vehicles stay in the platoon for the longest duration after the docking operation. This reduces energy consumption and improves mobility (Hall and Chin, 2005; Van De Hoef et al., 2015).

$$\text{FP}: \min_{T\in[0,\infty)} T.$$

The FP is subject to the feasibility of the second-stage trajectory optimization problem (SP) described below. The SP aims to determine the trajectory shapes to reduce the sum of squared acceleration, an indicator commonly used for riding comfort (Xu et al., 2019) and energy consumption (Y. Zhang et al., 2017), and maximize the travel distance, an indicator of mobility (Litman, 2003). This yields the following formulation:

$$\text{SP}: \min_{\{x_i(t)\}} \sum_{i=1}^{I} \int_0^T \ddot{x}_i^{\,2}(t)dt + c \times \sum_{i=1}^{I} \int_0^T [t \times \bar{v} - (x_i(t) - x_i(0))]dt,$$

subject to constraints (1)-(5). Note that in the second term of the objective, we reformulate maximizing the cumulative travel distance into minimizing the cumulative uncovered travel distance from each AMV's actual travel distance $(x_i(t) - x_i(0))$ to its ideal travel distance $t \times \bar{v}$ at the speed limit. This term is illustrated as the blue shaded area in Fig. 3. The reasons for such a formulation follow. First, it maximizes mobility along with the whole operation time horizon. Vehicles are encouraged to travel as fast as possible during the docking operation. This is valuable for improving upstream traffic mobility. For example, consider a case where another vehicle, traveling behind the operation platoon, intends to exit the current roadway. This vehicle could diverge sooner if the platoon operation mobility is maximized all the time. Second, this formulation pushes vehicles to platoon with their preceding vehicles as soon as possible, thus improving energy efficiency. Coefficient $c \in R^+$ denotes the weight of the accumulative uncovered travel distance relative to the sum of squared acceleration. For the convenience of the following analysis, let t^{d} denote the optimal solution to T.

It is worth noting that minimizing the operation time (FP) does not necessarily lead to substantial acceleration and deceleration in SP. Sizable speed adjustment can always be avoided by setting a smaller range of vehicle acceleration (i.e., $\left[\underline{a}, \bar{a}\right]$). Decreasing the acceleration range leads to an extended operation time and smoother trajectories.

The above-formulated problem is a trajectory planning problem that devises ideal trajectories to guide vehicles to form a tight platoon. Whereas, in real-world implementations, the planning problem is followed by a vehicle control problem, which regulates vehicle movements (e.g., throttle/brake) in the presence of various disturbances to follow the planned trajectories as much as possible. Various optimal control techniques have been proposed to reduce the control error (i.e., the difference between the vehicle actual trajectory and the planned target) such that the intended operation can be finished with expected performance. However, due to the unpredicted nature of the environment, it is possible that the control error is too big to be mitigated by the controller, or vehicles must decelerate hard to avoid collisions with certain objects. In real-world implementations, trajectory planning and vehicle control must be conducted in a rolling manner. In the beginning, vehicle trajectories are planned and followed in the presence of disturbances. When it is not feasible to follow the originally planned trajectories, new trajectories should be devised when appropriate and then followed, and so on until the intended operation is completed. The integrated trajectory planning and control has witnessed substantial development in robotics (De Luca and Oriolo, 2002; Liu et al., 2020) and has started attracting attention in the field of intelligent transportation systems (Turri et al., 2017; Wang et al., 2020). More advancements are expected in the future.

After the problem is formulated, the standard method first discretizes the model and then solves it with a solver. Yet, it is impossible to utilize an existing solver for the FP as it is subject to the feasibility of the SP. Thus, theoretical property analysis of the problem structure and solution structure is demanded. Even though the FP is solved, and the SP is formulated, it can take a rather long solution

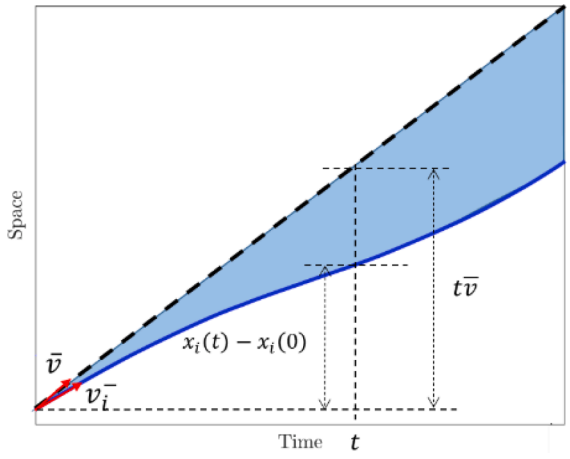


Fig. 3. Illustration of the accumulative uncovered travel distance.

time for the existing solver to solve the non-linear SP with differential equations as the problem size increases, i.e., the number of vehicles and the optimization time horizon length. In this case, the solution efficiency cannot meet the requirement of real-time vehicle control, which is usually executed at the sub-second level. Motivated by the above issues, we conduct theoretical property analysis in Section 3 that analytically solves the FP and adds valid cuts to the original feasible region to expedite the solution time of the existing solver. More importantly, theoretical properties enable the development of a fast heuristic (Section 4) that solves the SP significantly faster with an average solution time at the milli-second level without much loss of the solution optimality.

3. Theoretical properties

This section reveals the theoretical properties of the DP-TPP through feasibility analysis, which leads to an analytical solution to the FP. Also, the feasible region of the SP can be greatly reduced by introducing the global feasible cone obtained from feasibility analysis. These properties provide a foundation for the following heuristic and exact solution approaches. Section 3.1 revisits the definition of a set of trajectories merging operations from (Li and Li, 2019) as the basic terms for the following analysis. Section 3.2 investigates local feasibility for an individual AMV without considering conflicts with others. Section 3.3 analytically solves the FP by extending the time geography theory. Section 3.4 analyses the feasible region of the SP by considering other AMVs' impacts.

3.1. Merging operations

The following merging operations (Li and Li, 2019) are dedicated to smooth trajectories by avoiding kinks that violate the kinematic constraints.

3.1.1. Simple merging operation

The simple merging operation adds a transitional segment between two trajectory segments. The two trajectory segments are generated based on two different states (i.e., location, speed, time, and acceleration) of the same vehicle in the time–space diagram. This transitional segment enables the resulted trajectory to satisfy kinematic constraints (1) - (2).

Definition 1. *We define a quadratic trajectory $q_{svta}(t^{'})$ for a given original state (s, v, t) using an acceleration rate a as follows*:

$$q_{svta}(t^{'}) := s + v(t^{'} - t) + 0.5a(t^{'} - t)^2, \forall t^{'}.$$

All quadratic functions in the following sections should satisfy constraints (1) - (2), i.e., $\in [0, \bar{v}]$, $a \in \left[\underline{a}, \bar{a}\right]$.

Definition 2. *The simple merging operation between two quadratic trajectories is defined as follows. One trajectory $q_{s_1 v_1 t_1 a_1}$ (or q_1 for short) shoots forward from the initial point (s_1, v_1, t_1) with an acceleration rate of a_1. Another trajectory $q_{s_3 v_3 t_3 a_3}$ (or q_3 for short) shoots backward from the initial point (s_3, v_3, t_3) with an acceleration rate of a_3. The merging trajectory $q_{q_{s_3 v_3 t_3 a_3 (t2)} \dot{q}_{s_3 v_3 t_3 a_3 (t_2)} t_2 a_2}$ (or q_2 for short) shoots backward from its initial point, sliding along the trajectory q_3 at t_2 and merges with q_1 at t_m. As illustrated in Fig. 4, the merging trajectory q_2 is tangent to both trajectories q_1 and q_3. This indicates that the location and speed of q_1 at time t_m are equal to the location and speed of q_2 at time t_m. Also, the location and speed of q_2 at time t_2 are equal to the location and speed of q_3 at time t_2.*

With the above two conditions, starting time t_2 can be solved as follows:

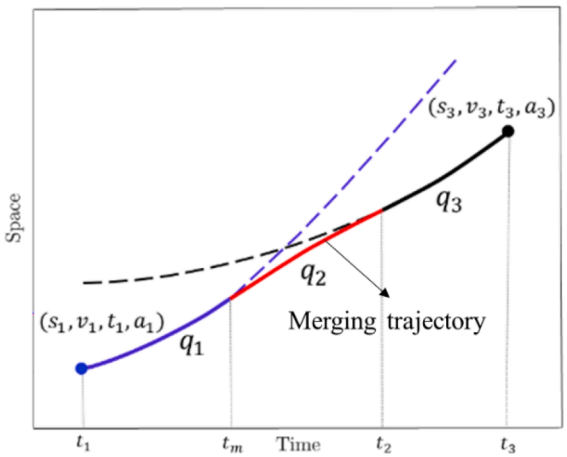


Fig. 4. An illustration of the simple merging operation.

$$t_2 = \begin{cases} -\frac{B}{2A} - \frac{\sqrt{B^2 - 4AC}}{2|A|}, if A > 0 and a_1 - a_3 > 0; \\ -\frac{B}{2A} + \frac{\sqrt{B^2 - 4AC}}{2|A|}, if A > 0 and a_1 - a_3 < 0; \\ -\frac{C}{B}, if A = 0, \end{cases}$$

$$\mathrm{A} = (a_1 - a_3)(a_2 - a_3),\ \mathrm{B} = 2(a_2 - a_3)(v_1 - v_3 - a_1 t_1 + a_3 t_3),\ \mathrm{C} = (v_1 - v_3 - a_1 t_1 + a_3 t_3)^2 - 2(a_1 - a_2)\left(\frac{a_1 {t_1}^2}{2} - \frac{a_3 {t_3}^2}{2} + t_3 v_3 - v_1 t_1 + s_1 - s_3\right)$$

Based on this, the merging point between q_1 and q_2, denoted by t_m, also can be analytically solved as follows:

$$t_m = \frac{v_3 - v_1 + a_1 t_1 - a_2 t_2 + a_3(t_2 - t_3)}{a_1 - a_2}.$$

3.1.1.1. Upper-bound merging operation. The upper-bound merging operation creates a trajectory satisfying kinematic constraints (1) and (2) as a tight upper bound to a set of trajectories. The resulting upper-bound trajectory will be used to construct the upper bound of a vehicle's feasible region to avoid collisions when considering the impacts from other vehicles.

Definition 3. *We define the upper-bound merging operation (UMO) for a series of trajectories $s := \{s_1, s_2, \cdots, s_L\}$ as follows.*

Step 0: Set s^{u} as an empty trajectory set and set starting time $t^- = 0$, the number of trajectory segments $k = 1$.

Step 1: Find the lowest trajectory segment from s at the right side of t^- and use s'_k to denote this trajectory segment, i.e. $s'_k(t) \le s_l(t)$, $\forall l = 1,2,\cdots,L, t \in [t^-, t^- + \delta], \exists \delta > 0$. Find $t^+ = sup_{t \in (t^-, \infty)} s'_k(t) \le s_l(t), \forall l = 1,2,\cdots,L$. If there is more than one trajectory segment, select the one with the largest t^+. Append $s'_k(t^- : t^+)$ to s^{u}, i.e.,$s^{\mathrm{u}} := [s^{\mathrm{u}}, s'_k(t^- : t^+)]$, and set ${t_k}^- = t^-, {t_k}^+ = t^+$.

Step 2: If $t^+ < \infty$, set $t^- = t^+$ and $k = k + 1$, and go to Step 1. Otherwise, if $t^+ = \infty$, set $K = k$, and go to Step 3.

Step 3: In the case of s^{u} has kinks that violate constraints (1) and (2), the simple merging operation is used to smooth these kinks. Set $k = 1$ and $k' = k + 1$.

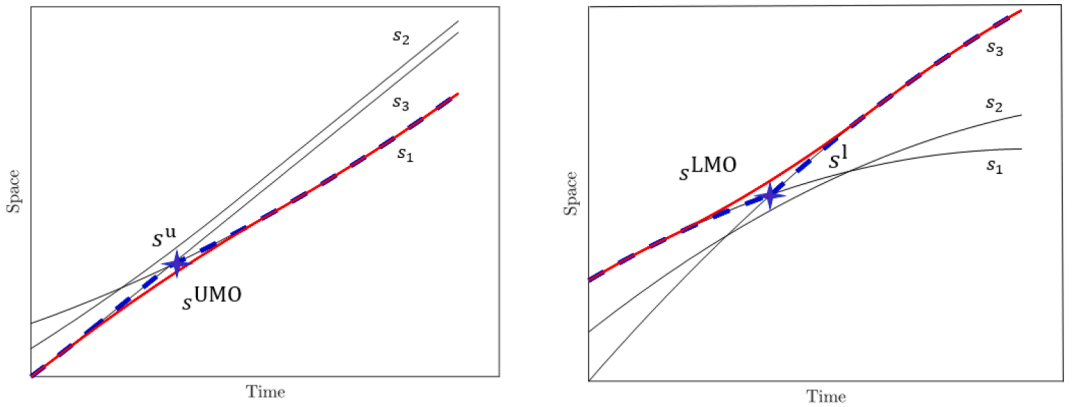
Step 4: Set s'_k as $q_1, s'_{k'}$ as q_3 and $\underline{a}$ as a_2 to solve the starting point t_2 and the merging point t_m. If $t_2 > {t_{k'}}^+$, replace q_3 with the next trajectory segment, i.e., set $k' = k' + 1$. If $t_m < {t_k}^-$, replace q_1 with the front trajectory segment, i.e., set $k = k - 1$. Repeat this step until $t_m \in [{t_k}^-, {t_k}^+]$ and $t_2 \in [{t_{k'}}^-, {t_{k'}}^+]$ are found.

Step 5: Use the smoothed segment $[s_k({t_k}^-, t_m), q_2(t_m, t_2), s_{k'}(t_2, {t_{k'}}^+)]$ to replace the unsmoothed segment $[s_k({t_k}^-, {t_k}^+), s_{k'}({t_{k'}}^-, {t_{k'}}^+)]$ in s^{u}. If $k' < K$, set $k = k'$ and $k' = k + 1$, go to Step 4. Otherwise, go to the next step.

Step 6: Return the trajectory of s after UMO, denoted by $s^{\mathrm{UMO}}(s) := s^{\mathrm{u}}$.

Fig. 5 (a) is an illustration of UMO for three trajectories $s = \{s_1, s_2, s_3\}$, shown in the black curves. We see that the blue dashed curve has a kink marked by the blue stars violating constraints (1) and (2). After UMO, s^{UMO} is smoothed and becomes the highest trajectory satisfying the aforementioned constraints and bounds these three trajectories from below.

3.1.1.2. Lower-bound merging operation. Opposite to the previous section, the lower-bound merging operation creates a trajectory satisfying kinematic constraints (1) and (2) as a tight lower bound to a set of trajectories. The resulting lower-bound trajectory and the



(a) UMO. (b) LMO.

Fig. 5. An illustration of UMO and LMO.

previous upper-bound trajectory will be used to construct an envelope of a vehicle's feasible region to avoid collisions when considering the impacts from other vehicles.

Definition 4. *The lower-bound merging operation (LMO) for a series of trajectories* $s := \{s_1, s_2, \cdots, s_L\}$ *is defined as follows. The major difference is that we use a maximum acceleration segment to do the simple merging operation.*

Step 0: Set $s^{\downarrow}$ as an empty trajectory set and set starting time $t^- = 0$, the number of trajectory segments $k = 1$.

Step 1: Find the highest trajectory segment from s on the right side of t^- and use s'_k to denote this segment, i.e. $s'_k(t) \le s_l(t), \forall l = 1, 2, \cdots, L, t \in [t^-, t^- + \delta], \exists \delta > 0$. Find $t^+ = sup_{t \in (t^-, \infty)} s'_k(t) \ge s_l(t), \forall l = 1, 2, \cdots, L$. If there is more than one trajectory segment, the one with the largest t^+ is selected. Append $s'_k(t^- : t^+)$ to $s^{\downarrow}$, i.e.,$s^{\downarrow} := [s^{\downarrow}, s'_k(t^- : t^+)]$, and set $t_k^- = t^-, t_k^+ = t^+$.

Step 2: If $t^+ < \infty$, set $t^- = t^+$ and $k = k + 1$, and go to Step 1. Otherwise, if $t^+ = \infty$, set $K = k$, and go to the next step.

Step 3: In the case of $s^{\downarrow}$ has kinks that violate constraints (1) and (2), the simple merging operation is used to smooth these kinks. Set $k = 1$ and $k' = k + 1$.

Step 4: Set s'_k as $q_1, s'_{k'}$ as q_3 and $\overline{a}$ as a_2 to solve the starting point t_2 and the merging point t_m. If $t_2 > t_{k'}{}^+$, replace q_3 with the next trajectory segment, i.e., set $k' = k' + 1$. If $t_m < t_k{}^-$, replace q_1 with the front trajectory segment, i.e., set $k = k - 1$. Repeat this step until $t_m \in [t_k{}^-, t_k{}^+]$ and $t_2 \in [t_{k'}{}^-, t_{k'}{}^+]$ are found.

Step 5: Use the smoothed segment $[s_k(t_k{}^-, t_m), q_2(t_m, t_2), s_{k'}(t_2, t_{k'}{}^+)]$ to replace the unsmoothed segment $[s_k(t_k{}^-, t_k{}^+), s_{k'}(t_{k'}{}^-, t_{k'}{}^+)]$ in $s^{\downarrow}$. If $k' < K$, set $k = k'$ and $k' = k + 1$, go to Step 4. Otherwise, go to the next step.

Step 6: Return the trajectory of s after LMO, denoted by $s^{\mathrm{LMO}}(s) := s^{\downarrow}$.

Fig. 5 (b) is an illustration of LMO for another set of trajectories $s = \{s_1, s_2, s_3\}$. Again, we see that $s^{\downarrow}$, shown in the blue dashed curve, may have kinks violating the kinematic constraints before smoothing. But after the LMO, s^{LMO} satisfies the kinematic constraints and bounds these trajectories from above.

3.2. Local feasibility analysis

The local feasibility analysis pertains to an individual AMV without considering impacts from other AMVs (i.e., only subject to constraints (1), (2), and (4) without considering constraints (3) for conflicts of multiple vehicles).

3.2.1. Local feasible cone

Definition 5. *We define a local feasible cone for each AMV in the time–space diagram as the region between the maximum acceleration trajectory and the maximum deceleration trajectory. The maximum acceleration trajectory is the furthest trajectory that the AMV can reach subject to its initial boundary conditions (4) without exceeding the speed and acceleration upper limits (i.e.,* $\overline{v}$ *and* $\overline{a}$*). In contrast, the maximum deceleration trajectory is the nearest trajectory that the AMV can maintain subject to its initial boundary conditions (4) without exceeding the speed and acceleration lower limits (i.e.,* 0 *and* $\underline{a}$ *). With this definition, all feasible trajectories for the AMV will be always within the local feasible cone.*

Let $\overline{s}_{s_i v_i t_i a_i}$ ($\overline{s}_i$ for short) denote the acceleration trajectory of AMV i that starts at state $(s_i v_i t_i)$ with a positive acceleration rate $(a_i > 0)$ and $\underline{s}_{s_i v_i t_i a_i}$ ($\underline{s}_i$ for short) denote the deceleration trajectory that starts at state $(s_i v_i t_i)$ with the negative acceleration rate $(a_i < 0)$. With the positive acceleration rate, the trajectory will accelerate to the speed $\overline{v}$ and then cruise at the maximum speed. If the acceleration rate is negative, the trajectory will decelerate to speed 0 and stop for the rest of the time. $\overline{s}_{s_i v_i t_i a_i}(t)$ and $\underline{s}_{s_i v_i t_i a_i}(t)$ can be formulated as follows:

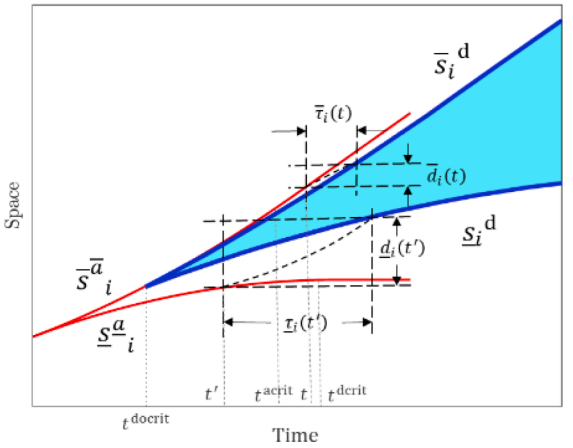


Fig. 6. Illustration of a local feasible cone.

$$\overline{s}_{s_i v_i t_i a_i}(t) = \begin{cases} s_i + v_i(t-t_i) + 0.5a_i(t-t_i)^2, \forall t \in \left[t_i, t_i^{\text{acrit}}\right], \\ \overline{s}_{s_i v_i t_i a_i}\left(t_i^{\text{acrit}}\right) + \overline{v}\left(t - t_i^{\text{acrit}}\right), \forall t > t_i^{\text{acrit}}, \end{cases} if a_i \geq 0;$$

$$\underline{s}_{s_i v_i t_i a_i}(t) = \begin{cases} s_i + v_i(t-t_i) + 0.5a_i(t-t_i)^2, \forall t \in \left[t_i, t_i^{\text{dcrit}}\right], \\ \underline{s}_{s_i v_i t_i a_i}\left(t_i^{\text{dcrit}}\right), \forall t > t_i^{\text{dcrit}}, \end{cases} if a_i < 0, \quad (6)$$

where the critical time between the quadratic acceleration segment and the linear segment is defined as:

$$t_i^{\text{acrit}} := t_i + \frac{\overline{v} - v_i}{a_i}, \text{ if} a_i \geq 0.$$

And the critical time between the quadratic deceleration segment and the linear segment is defined as:

$$t_i^{\text{dcrit}} := t_i + \frac{-v_i}{a_i}, \text{if} a_i < 0.$$

When a_i equals $\overline{a}$ or $\underline{a}$, the above equation yields the maximum acceleration trajectory $\overline{s}_{s_i v_i t_i \overline{a}}$ ($\overline{s^a}_i$for short) or the maximum deceleration trajectory $\underline{s}_{s_i v_i t_i \underline{a}}$ ($\underline{s}^{\underline{a}}{}_i$for short). The region between these two trajectories in red thin curves in Fig. 6 is the local feasible cone of the AMV.

3.2.2. Local docking feasible cone

Definition 6. *We define a local docking feasible cone for each AMV in the time–space diagram as the region where this AMV can reach the docking speed. The local feasible cone specifies the region for all possible space–time points for the AMV to complete the docking operation with the neighboring AMV(s). It can be obtained based on the corresponding local feasible cone.*

We let each point on the upper bound of local feasible cone $\overline{s^a}_i$ with a speed greater than v^d decelerate at maximum deceleration rate $\underline{a}$ until it reaches v^d, which yields the upper bound of the docking feasible cone $\overline{s}_i{}^{\text{d}}$. Similarly, let each point on lower bound $\underline{s}^{\underline{a}}{}_i$ with a speed less than v^d accelerate at the maximum acceleration rate $\overline{a}$ until it reaches v^d , which yields us the lower bound of the docking feasible cone $\underline{s}_i{}^{\text{d}}$. Fig. 6 illustrates a local docking feasible cone, shown in blue bold curves when the original speed is less than the docking speed. The key parameters of the local docking feasible cone are formulated as follows. $\overline{\tau}_i(t)$ denotes the time duration needed to decelerate from the point on $\overline{s^a}_i$ at time instant t to v^d. $\dot{\overline{s^a}}_i$ is the first order derivative of the position $\overline{s^a}_i$ and it denotes the corresponding speed. t_i^{docrit} is the earliest time point that AMV i reaches docking speed v^d.

$$t_i^{\text{docrit}} = \frac{v^{\text{d}} - \dot{\overline{s^a}}_i(0)}{\overline{a}}.$$

$$\overline{\tau}_i(t) = \frac{v^{\text{d}} - \dot{\overline{s^a}}_i(t)}{\underline{a}}, \forall t \geq t_i^{\text{docrit}}.$$

$\overline{d}_i(t)$ denotes the distance traveled to decelerate from the point on $\overline{s^a}_i$ at time instant t to v^d.

$$\overline{d}_i(t) = \frac{(v^{\text{d}})^2 - (\dot{\overline{s^a}}_i(t))^2}{2\underline{a}}, \forall t \geq t_i^{\text{docrit}}.$$

$\overline{s}_i{}^{\text{d}}$ denotes the upper bound to the local docking feasible cone.

$$\overline{s}_i{}^{\text{d}}(t + \overline{\tau}_i(t)) = \overline{s^a}_i(t) + \overline{d}_i(t), \forall t \geq t_i^{\text{docrit}}. \quad (7)$$

$\underline{\tau}_i(t')$ denotes the time duration needed to accelerate from the point on $\underline{s}^{\underline{a}}{}_i$ at time instant t' to v^d. $\dot{\underline{s}}^{\underline{a}}{}_i$ is the first-order derivative of the position $\underline{s}^{\underline{a}}{}_i$ and it denotes the corresponding speed.

$$\underline{\tau}_i(t') = \frac{v^{\text{d}} - \dot{\underline{s}}^{\underline{a}}{}_i(t')}{\overline{a}}, \forall t' \geq 0.$$

$\underline{d}_i(t')$ denotes the distance traveled to accelerate from the point on $\underline{s}^{\underline{a}}{}_i$ at time instant t' to v^d.

$$\underline{d}_i(t') = \frac{(v^{\text{d}})^2 - \left(\dot{\underline{s}}^{\underline{a}}{}_i(t')\right)^2}{2\overline{a}}, \forall t' \geq 0.$$

$\underline{s}_i{}^{\text{d}}$ denotes the lower bound to the local docking feasible cone.

$$\underline{s}_i{}^{\text{d}}\left(t' + \underline{\tau}_i(t')\right) = \underline{s}^{\underline{a}}{}_i(t') + \underline{d}_i(t'), \forall t' \geq 0. \quad (8)$$

3.3. Minimum docking time

This section shows that the above feasibility cones yield a lower bound to the docking time T, i.e., the optimal solution to the FP.

Definition 7. *We define a local docking feasible shadow cone* DSC_i *for AMV i in the time–space diagram. It is obtained by moving AMV i's docking feasible cone downwards by the cumulative distance of its upstream AMV lengths.* DSC_i *is defined to reduce each AMV as a dimensionless point by eliminating the vehicle length. The upper bound and lower bound are denoted by* $\overline{\mathrm{DSC}_i}$ *(the red dashed curve in Fig. 7) and* $\underline{\mathrm{DSC}_i}$ *(the blue solid curve in Fig. 7).*

$$\overline{\mathrm{DSC}_i} = \overline{s}_i^{\,\mathrm{d}} - \sum_{j=i}^{I-1} l_j,\ \forall i \in \mathscr{I}.$$

$$\underline{\mathrm{DSC}_i} = \underline{s}_i^{\,\mathrm{d}} - \sum_{j=i}^{I-1} l_j, \forall i \in \mathscr{I}. \tag{9}$$

Definition 8. *We define an intersection point denoted by* $\mathrm{X}_{jk}, \forall j,k \in \mathscr{I}$ and $j<k$, *between the upper bound* $\overline{\mathrm{DSC}_j}$ *of vehicle j's local docking feasible shadow cone and the lower bound* $\underline{\mathrm{DSC}_k}$ *of vehicle k's local docking feasible shadow cone in the time–space diagram, as marked in the green dots in Fig. 7.* X_{jk} *is formulated as*:

$$\mathrm{X}_{jk} := \overline{\mathrm{DSC}_j} \cap \underline{\mathrm{DSC}_k}, j,k \in \mathscr{I} \text{ and } j>k. \tag{10}$$

The minimum docking time t^{d} is obtained as the time value of the rightmost intersection point, marked as the black star in Fig. 7.

$$t^{\mathrm{d}} := \max_{j,k\in\mathscr{I} \text{ and } j>k} \left\{t|(t,x) \in \mathrm{X}_{jk}\right\}. \tag{11}$$

The two critical AMV indexes j and k are denoted as i^* and i'.

$$\{i^*, i'\} := \underset{j,k\in\mathscr{I} \text{ and } j>k}{\operatorname{argmax}} \left\{t|(t,x) \in \mathrm{X}_{jk}\right\}. \tag{12}$$

Theorem 1. *$t^{\mathrm{d}}$is the minimum feasible docking time for the DP-TPP.*

Proof: We can assume that a docking time t' smaller than t^{d} exists, as shown in Fig. 7. There is no common area of all AMVs' local docking feasible shadow cones, indicating that not all AMVs can dock together at t'. Thus, a docking time smaller than t^{d} does not exist. Later analysis (see Theorem 3 in Section 4) shows that t^{d} is feasible to the DP-TPP. This completes the proof.

Theorem 2. *When $T = t^{\mathrm{d}}$, the final locations of all AMVs are fixed.*

Proof: t^{d} is the time value of the rightmost intersection point $\mathrm{X}_{i^*i'}$. Thus, when $T = t^{\mathrm{d}}$, AMV i' will decelerate the most, stop if needed, and then accelerate the most to reach v^{d} by T. The final location of AMV i' is:

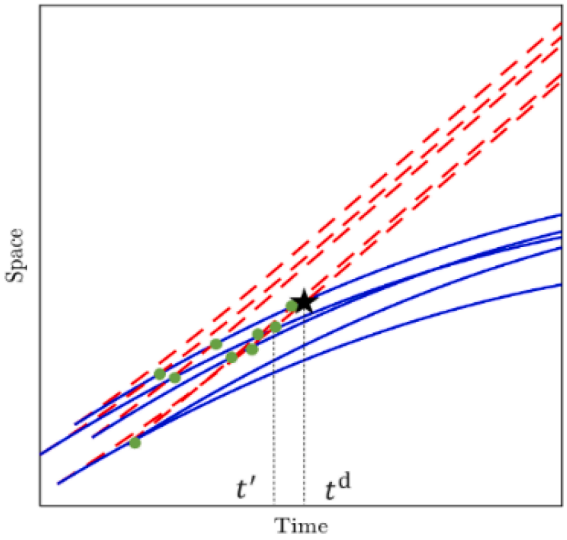


Fig. 7. Illustration of the minimum docking time.

$$x_{i^-}(t^{\mathrm{d}}) = \begin{cases} s_{i^-} + v_{i^-} t^{\mathrm{c}} + \frac{1}{2}\underline{a} \times (t^{\mathrm{c}})^2 + \left(v_{i^-} + \underline{a}\, t^{\mathrm{c}}\right) \times \left(t^{\mathrm{d}} - t^{\mathrm{c}}\right) + \frac{1}{2}\overline{a} \times \left(t^{\mathrm{d}} - t^{\mathrm{c}}\right)^2, if\, v_{i^-} \geq \frac{\underline{a}\, v^{\mathrm{d}} - \underline{a}\, \overline{a} t^{\mathrm{d}}}{\overline{a}}; \\ s_{i^-} + v_{i^-} t^{\mathrm{c1}} + \frac{1}{2}\underline{a} \times \left(t^{\mathrm{c1}}\right)^2 + \overline{v} \times \left(t^{\mathrm{c2}} - t^{\mathrm{c1}}\right) + \overline{v} \times \left(t^{\mathrm{d}} - t^{\mathrm{c2}}\right) + \frac{1}{2}\overline{a} \times \left(t^{\mathrm{d}} - t^{\mathrm{c2}}\right)^2, otherwise. \end{cases}$$

where $t^{\mathrm{c}} = \frac{v^{\mathrm{d}} - v_{i^-}^- - \overline{a} t^{\mathrm{d}}}{\underline{a} - \overline{a}}$, $t^{\mathrm{c1}} = \frac{-v_{i^-}^-}{\underline{a}}$, and $t^{\mathrm{c2}} = t^{\mathrm{d}} - \frac{v^{\mathrm{d}}}{\overline{a}}$.

Then all AMVs' final locations can be formulated as follows:

$$x_i(t^{\mathrm{d}}) = \begin{cases} x_{i^*}(t^{\mathrm{d}}) + (i^* - i) \times l, if i < i^*; \\ x_{i^*}(t^{\mathrm{d}}), if i = i^*; \\ x_{i^*}(t^{\mathrm{d}}) - (i - i^*) \times l, if i > i^*. \end{cases} \tag{13}$$

This completes the proof.

3.4. Feasible region analysis

The feasible region analysis enables the development of a parsimonious trajectory planning approach considering other AMVs instead of only focusing on the object AMV. It also helps improve the solution efficiency of the discrete-time optimization model by greatly cutting the feasible region.

3.4.1. Quasi-global feasible cone

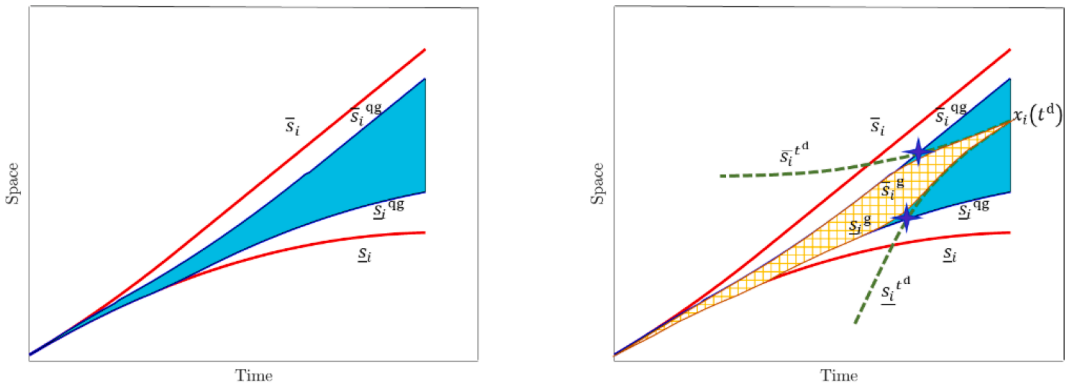
Definition 9. *A quasi-global feasible cone for AMV i is defined in the time–space diagram, which is based on its local feasible cone but considering the impact from other AMVs' local feasible cones. Apply UMO to a set of bounds* $s := \{\overline{s}_j - (i-j)^* l\}_{j\in[1,i]}$ *to get the upper bound of AMV i's quasi-global feasible cone, denoted by* $\overline{s}_i^{\mathrm{qg}}$. *Similarly, use LMO to another set of bounds* $s := \left\{\underline{s}_j + (j-i)^* l\right\}_{j\in[i,I]}$ *to get the lower bound of AMV i's quasi-global feasible cone, denoted by* $\underline{s}_i^{\mathrm{qg}}$. Fig. 8 (a) *illustrates a quasi-global feasible cone shown in the blue region.*

3.4.2. Global feasible cone

Definition 10. *A global feasible cone for each AMV is defined in the time–space diagram that further cuts the feasible region compared with the above quasi-global feasible cone. One acceleration trajectory* $\overline{s}_i^{t^{\mathrm{d}}}$ *using the maximum acceleration rate* $\overline{a}$ *and one deceleration trajectory* $\underline{s}_i^{\ t^{\mathrm{d}}}$ *using the maximum deceleration rate* $\underline{a}$ *are anchored at AMV i's final location* $x_i(t^{\mathrm{d}})$ *with docking speed (i.e.,* $\dot{x}_i(t^{\mathrm{d}}) = v^{\mathrm{d}}$*) and shooting backward, shown in* Fig. 8 (b). *Kinks exist at the intersection of the quasi-global feasible cone upper bound* $\overline{s}_i^{\mathrm{qg}}$ *(or lower bound* $\underline{s}_i^{\mathrm{qg}}$*) and the acceleration trajectory* $\overline{s}_i^{t^{\mathrm{d}}}$ *(or deceleration trajectory* $\underline{s}_i^{\ t^{\mathrm{d}}}$*), marked as blue star in* Fig. 8 (b). *Thus, the simple merging operation is used to merge* $\overline{s}_i^{\mathrm{qg}}$ *(or* $\underline{s}_i^{\mathrm{qg}}$*) and* $\overline{s}_i^{t^{\mathrm{d}}}$ *(or* $\underline{s}_i^{\ t^{\mathrm{d}}}$*) to yield a smoother global feasible cone upper bound* $\overline{s}_i^{\mathrm{g}}$ *(or lower bound* $\underline{s}_i^{\mathrm{g}}$*), shown in* Fig. 8 (b). *The yellow grid region between* $\overline{s}_i^{\mathrm{g}}$ *and* $\underline{s}_i^{\mathrm{g}}$ *is AMV i's global feasible cone.*

It is noteworthy that only regions in the time–space diagram where vehicles cannot reach due to their kinematic limits and collision avoidance needs are cut out. Thus, all feasible solutions, including the optimal solution, must reside in the global feasible region.

If initial boundary conditions $\{s_i, v_i\}$ are not properly set, $\overline{s}_i^{\mathrm{g}}$ may not always be above $\underline{s}_i^{\mathrm{g}}$. Thus, the global feasible cone could be empty (or the DP-TPP instance does not have a feasible solution). We call a set of boundary conditions that yields $\overline{s}_i^{\mathrm{g}}(t) \geq \underline{s}_i^{\mathrm{g}}(t), \forall t \in$



(a) Quasi-global feasible cone. (b) Global feasible cone.

Fig. 8. Illustrations of feasible cones.

$[0, T]$ "admissible". If the boundary conditions are inadmissible, the AMVs will collide inevitably, and it is impossible to complete the docking operation. Thus, this paper only considers admissible boundary conditions. For further explanation about admissible boundary conditions, see Theorem 4 in Section 4.

4. Parsimonious heuristic trajectory planning

The theoretical property analysis enables the development of a parsimonious heuristic trajectory planning algorithm solving the DP-TPP that obtains near-optimal trajectories as a small number of simple quadratic segments and significantly improves solution efficiency compared with commercial optimization solver. The basic idea of the algorithm is to let AMV i^* follow its global feasible cone lower bound. For AMVs ahead of i^*, we plan vehicle trajectories from upstream to downstream by letting each of them merge into the following vehicle's shadow trajectory. For AMVs behind i^*, we plan vehicle trajectories from downstream to upstream by letting each of them merge into the preceding vehicle's shadow trajectory. Further, a parsimonious heuristic alternative solution that obtains the exact optimal trajectories is proposed based on Pontryagin's maximum principle to solve a special case of the DP-TPP with safety constraints relaxed.

4.1. Algorithm description

We let AMV i^* follow its global feasible cone lower bound $\underline{s}_{i^*}^{\mathrm{g}}$, and its trajectory is formulated as follows:

$$X_{i^*}^{\mathrm{P,rd}}(t) = \underline{s}_{i^*}^{\mathrm{g}} \tag{14}$$

Definition 11. *We define an upstream shadow trajectory and a downstream shadow trajectory of $X_i(t)$, respectively.*

$$X_i^{\mathrm{s-}}(t) = X_i^{\mathrm{P,rd}}(t) - l_i \forall i, t,$$

$$X_i^{\mathrm{s+}}(t) = X_i^{\mathrm{P,rd}}(t) + l_i \forall i, t,$$

where l_i is the AMV length.

For AMVs behind i^*, vehicle trajectories are solved from downstream to upstream. AMV i, $i > i^*$, follows the upper bound $\bar{s}_{s_i v_i t_i a^{\mathrm{fw}}{}_i}$ (short as $\bar{s}_i$) with a certain acceleration $a^{\mathrm{fw}}{}_i$ to intersect with the preceding AMV's upstream shadow trajectory $X_{i-1}{}^{\mathrm{s-}}$. This way, the trajectory may not satisfy acceleration constraints (2). Thus, the simple merging operation is used to smooth the trajectory. When $i > i^*$, the trajectory of one AMV can be solved with **Algorithm 1**.

To smooth vehicle trajectories, a binary search is conducted to find the minimum feasible acceleration rate $a^{\mathrm{fw}}{}_i, i > i^*$, for each upstream AMV after i^*'s trajectory is obtained based on equation (14). All upstream AMVs' trajectories can be solved with **Algorithm 2**, a parsimonious heuristic approach.

Algorithm 1. The algorithm for solving the trajectory of one upstream AMV,$i > i^*$

Step 0: If $i = i^* + 1$, go to Step 1; otherwise, go to Step 2.

Step 1: If $t^{\mathrm{c}} \le t_{i^*}^{\mathrm{dcrit}}$, set $\mathscr{T} = \{0, t^{\mathrm{c}}, t^{\mathrm{d}}\}, \mathscr{A} = \left\{\underline{a}, \bar{a}, 0\right\}, \mathscr{S} = \left\{X_{i^*}{}^{\mathrm{s-}}(0), X_{i^*}{}^{\mathrm{s-}}(t^{\mathrm{c}}), X_{i^*}{}^{\mathrm{s-}}(t^{\mathrm{d}})\right\}$, and $\mathscr{V} = \left\{\dot{X}_{i^*}(0), \dot{X}_{i^*}(t^{\mathrm{c}}), v^{\mathrm{d}}\right\}$; otherwise, set $\mathscr{T} = \{0, t_{i^*}^{\mathrm{dcrit}}, t^{\mathrm{c}}, t^{\mathrm{d}}\}$, $\mathscr{A} = \left\{\underline{a}, 0, \bar{a}, 0\right\}, \mathscr{S} = \left\{X_{i^*}{}^{\mathrm{s-}}(0), X_{i^*}{}^{\mathrm{s-}}(t_{i^*}^{\mathrm{dcrit}}), X_{i^*}{}^{\mathrm{s-}}(t^{\mathrm{c}}), X_{i^*}{}^{\mathrm{s-}}(t^{\mathrm{d}})\right\}$ and $\mathscr{V} = \{\dot{X}_{i^*}(0), 0, 0, v^{\mathrm{d}}\}$. $j = 1, j \in \mathscr{J}\backslash\{J\}, \mathscr{J} = \mathit{length}(\mathscr{T})$.

Step 2: First, we assume $\bar{s}_i(t)$ is a quadratic trajectory and conduct the merging operation with $\bar{s}_{s_i v_i t_i a^{\mathrm{fw}}{}_i}(t)$ being $q_1(t)$, $X_{i-1}{}^{\mathrm{s-}}(t)$ being $q_3(t)$ ($a_3 = \mathscr{A}\{j\}, v_3 = \mathscr{V}\{j\}$, $t_3 = \mathscr{T}\{j\}$ and$s_3 = \mathscr{S}\{j\}$) and $-a^{\mathrm{fw}}{}_i$ being a_2 when $t \in [\mathscr{T}\{j\}, \mathscr{T}\{j+1\}]$. Solve the starting point t_2 and merging point t_m. We use t_{i_2} and t_{i_m} to denote the starting point t_2 and merging point t_m of AMV i.

Step 3: If $t_{i\_2} \in [\mathscr{T}\{j\}, \mathscr{T}\{j+1\}]$ and $t_{i\_m} \le t_i^{\mathrm{acrit}}$ go to Step 4; otherwise, set $j = j + 1$, and go to Step 2 until $j = J-1$. When $j = J-1$, if $t_{i\_2} \in [\mathscr{T}\{j\}, \mathscr{T}\{j+1\}]$ and $t_{i\_m} \le t_i^{\mathrm{acrit}}$ go to Step 4. Otherwise, go to Step 5.

Step 4: Set $\mathscr{T}^* = \{t_i, t_{i\_m}, t_{i\_2}\}, \mathscr{T}^{**} = \mathscr{T}\{j+1 : \mathit{length}(\mathscr{T})\}; \mathscr{A}^* = \left\{a^{\mathrm{fw}}{}_i, -a^{\mathrm{fw}}{}_i, \mathscr{A}(j)\right\}, \mathscr{A}^{**} = \mathscr{A}\{j+1 : \mathit{length}(\mathscr{A})\}; \mathscr{S}^* = \{s_i, \bar{s}_i(t_{i\_m}), X_{i-1}{}^{\mathrm{s-}}(t_{i\_2})\}, \mathscr{S}^{**} = \mathscr{S}\{j+1 : \mathit{length}(\mathscr{S})\}; \mathscr{V}^* = \left\{v_i, \bar{s}_i(t_{i\_m}), \dot{X}_{i-1}{}^{\mathrm{s-}}(t_{i\_2})\right\}, \mathscr{V}^{**} = \mathscr{V}\{j+1 : \mathit{length}(\mathscr{V})\}$. And update $\mathscr{T} = [\mathscr{T}^*, \mathscr{T}^{**}], \mathscr{A} = [\mathscr{A}^*, \mathscr{A}^{**}], \mathscr{S} = [\mathscr{S}^*, \mathscr{S}^{**}] - l, \mathscr{V} = [\mathscr{V}^*, \mathscr{V}^{**}]$. Go to Step 8.

Step 5: Now, we assume that $\bar{s}_i(t)$ is a compound function and reset $j = 1, j \in \mathscr{J}\backslash\{J\}, \mathscr{J} = \mathit{length}(\mathscr{T})$.

Step 6: We conduct the merging operation with the linear part of $\bar{s}_i(t)$ being $q_1(t)$, $X_{i-1}{}^{\mathrm{s-}}(t)$ being $q_3(t)$ ($a_3 = \mathscr{A}\{j\}, v_3 = \mathscr{V}\{j\}, t_3 = \mathscr{T}\{j\}$ and$s_3 = \mathscr{S}\{j\}$) and $-a^{\mathrm{fw}}{}_i$ being a_2 when $t \in [\mathscr{T}\{j\}, \mathscr{T}\{j+1\}]$. Solve the starting point t_{i_2} and merging point t_{i_m}.

Step 7: If $t_{i\_2} \in [\mathscr{T}\{j\}, \mathscr{T}\{j+1\}]$ and $t_{i\_m} \ge t_i^{\mathrm{acrit}}$, set $\mathscr{T}^* = \{t_i, t_i^{\mathrm{acrit}}, t_{i\_m}, t_{i\_2}\}, \mathscr{T}^{**} = \mathscr{T}\{j+1 : \mathit{length}(\mathscr{T})\}; \mathscr{A}^* = \left\{a^{\mathrm{fw}}{}_i, 0, -a^{\mathrm{fw}}{}_i, \mathscr{A}(j)\right\}, \mathscr{A}^{**} = \mathscr{A}\{j+1 : \mathit{length}(\mathscr{A})\}; \mathscr{S}^* = \left\{s_i, \bar{s}_i(t_i^{\mathrm{acrit}}), \bar{s}_i(t_{i\_m}), X_{i-1}{}^{\mathrm{s-}}(t_{i\_2})\right\}, \mathscr{S}^{**} = \mathscr{S}\{j+1 : \mathit{length}(\mathscr{S})\}; \mathscr{V}^* = \left\{v_i, \bar{v}, \bar{v}, \dot{X}_{i-1}(t_{i\_2})\right\}, \mathscr{V}^{**} = \mathscr{V}\{j+1 : \mathit{length}(\mathscr{V})\}$. And update $\mathscr{T} = [\mathscr{T}^*, \mathscr{T}^{**}]$, $\mathscr{A} = [\mathscr{A}^*, \mathscr{A}^{**}], \mathscr{S} = [\mathscr{S}^*, \mathscr{S}^{**}] - l, \mathscr{V} = [\mathscr{V}^*, \mathscr{V}^{**}]$. Go to Step 8; otherwise, set $j = j + 1$, and go to Step 6 until $j = J-1$.

Step 8: AMV i's trajectory $X_i^{\mathrm{P,rd}}, i > i^*$, can be formulated as:

$$X_i^{\mathrm{P,rd}}(t) = \begin{cases} \bar{s}_i(t), \forall t \in [0,\ t_{i\_m}], \\ \bar{s}_i(t_{i\_m}) + \dot{\bar{s}}_i(t_{i\_m}) \times (t - t_{i_m}) - 0.5 a^{\mathrm{fw}}{}_i \times (t - t_{i\_m})^2, \forall t \in [t_{i\_m}, t_{i\_2}], \\ X_{i-1}{}^{\mathrm{s-}}(t), \forall t \in [t_{i\_2}, T]. \end{cases} \tag{15}$$

For AMVs ahead of i^*, vehicle trajectories are solved from upstream to downstream. AMV i, $i < i^*$, follows the lower bound $\underline{s}_{s_i v_i t_i(-a^{\mathrm{fw}}{}_i)}$ (short as $\underline{s}_i$) with a certain acceleration $-a^{\mathrm{fw}}{}_i$ to intersect with the following AMV's downstream shadow trajectory $X_{i+1}{}^{\mathrm{s}+}$. The algorithm follows the same procedure as in **Algorithm 2** and thus is omitted here.

The PH is constructed so that both the squared acceleration and the mobility terms in the SP are accounted for. Specifically, the binary search of the acceleration rate makes sure that the value of the square acceleration term is not too large. Further, letting a vehicle follow the acceleration trajectory (or the deceleration trajectory) to merge into its preceding vehicle's shadow trajectory (or its following vehicle's shadow trajectory) ensures that mobility is considered. While it is difficult to theoretically prove the optimality performance of the PH in solving platooning trajectories given the numerical nature of the binary search, extensive numerical experiments with a wide range of key parameter settings are conducted in Section 6 to compare the solution performance of the PH with that of the solver. For all the instances tested, the PH solutions have a near-zero optimality gap, which validates the quality of the PH solutions.

4.2. Theoretical properties

This section analyzes the feasibility and optimality properties of the proposed PH algorithm. Theorem 3-Theorem 5 show that the feasibility of the PH output is equivalent to the feasibility of the DP-TPP. The following analysis proposes a heuristic alternative solution based on Pontryagin's maximum principle to solve a special case of the DP-TPP.

Theorem 3. *If the DP-TPP is feasible, $T = t^{\mathrm{d}}$ exists, and trajectories $X_i^{\mathrm{P}\cdot t^{\mathrm{d}}}$ and $\left\{X_i^{\mathrm{P}\cdot t^{\mathrm{d}}}\right\}_{i\in\mathcal{I}\setminus\{i^*\}}$ obtained from **Algorithm 1** with $a^{\mathrm{fw}}{}_i = \bar{a}$, $i\in\mathcal{I}\setminus\{i^*\}$ exist and are feasible to constraints (1) - (5).*

Proof: When $T = t^{\mathrm{d}}$, shown as the black star in Fig. 9, AMV i^* follows its global feasible cone lower bound. The shadow trajectory of $X_{i^*}^{\mathrm{P}\cdot t^{\mathrm{d}}}$ is shown in the green bold dashed curve in Fig. 9. When $i > i^*$, the shadow of AMV i's local feasible cone upper bound $\bar{s}_{s_i v_i t_i \bar{a}}$ will intersect with its preceding vehicle's shadow trajectory before time t^{d}. An illustrative intersection is marked as the black dot. This means each upstream AMV can dock with its preceding AMV if it follows the local feasible cone upper bound; i.e., $\{X_i^{\mathrm{P}\cdot t^{\mathrm{d}}}\}_{i>i^*}$ exist. Similarly, when $i < i^*$, the shadow of AMV i's local feasible cone lower bound $\underline{s}_{s_i v_i t_i \underline{a}}$ will intersect with its following vehicle's shadow trajectory before time t^{d}. An illustrative intersection is marked as the black square. This means each downstream AMV can dock with its following AMV if it follows the local feasible cone lower bound; i.e., $\{X_i^{\mathrm{P}\cdot t^{\mathrm{d}}}\}_{i<i^*}$ exist.

AMV i^*'s trajectory is its global feasible cone lower bound $\underline{s}_{i^*}{}^{\mathrm{g}}$ that must satisfy constraints (1) - (5) according to Definition 10. When $i > i^*$ AMV i' speed can be derived from equation (13) as:

$$\dot{X}_i^{\mathrm{P}\cdot t^{\mathrm{d}}}(t) = \begin{cases} \dot{\bar{s}}_i(t), \forall t \in [0,\ t_{i\_m}], \\ \dot{\bar{s}}_i(t_{i\_m}) - \bar{a}t, \forall t \in [t_{i\_m}, t_{i\_2}], \\ \dot{X}_{i-1}{}^{\mathrm{s}}(t), \forall t \in [t_{i\_2}, T]. \end{cases} \tag{16}$$

And the acceleration is derived as:

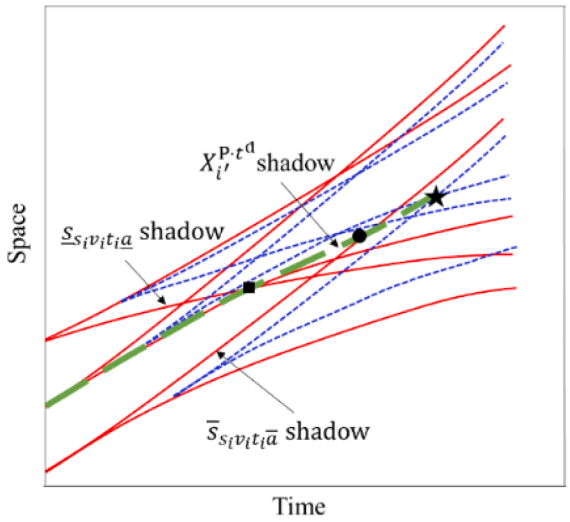


Fig. 9. An illustration of the minimum docking time feasibility.

$$\ddot{X}_i^{\mathrm{P}\cdot t^{\mathrm{d}}}(t)=\begin{cases}\overline{a},\forall t\in[0,\ t_{i\_m}],\\ -\overline{a}_i,\forall t\in[t_{i\_m},t_{i\_2}],\\ \ddot{X}_{i-1}^{\ s}(t),\forall t\in[t_{i\_2},T].\end{cases} \tag{17}$$

According to equations (16) and (17), $X_{i'+1}^{\mathrm{P}\cdot t^{\mathrm{d}}}$ first follows AMV $i'+1$'s local feasible cone upper bound $\overline{s}_{i'+1}^{\ \overline{a}}$ and then merges into $X_{i'}^{\mathrm{P}\cdot t^{\mathrm{d}}}$ shadow. Therefore, $X_{i'+1}^{\mathrm{P}\cdot t^{\mathrm{d}}}$ satisfies constraints (1) - (5). Follow this logic, we can conclude that $X_i^{\mathrm{P}\cdot t^{\mathrm{d}}}, i>i'$, follows AMV i's local feasible cone upper bound $\overline{s}_i^{\overline{a}}$ and then merges into $X_{i-1}^{\mathrm{P}\cdot t^{\mathrm{d}}}$ shadow. The same conclusion can be drawn when $i<i'$. Thus, $X_i^{\mathrm{P}\cdot t^{\mathrm{d}}}, \forall i\in\mathscr{I}$ satisfies constraints (1) - (5). This completes the proof.

Theorem 4. *The DP-TPP is feasible if and only if the initial boundary conditions are admissible.*

Proof: For the necessity, when AMV i's global feasible cone lower bound $\underline{s}_i^{\mathrm{g}}$ (green thin curve in Fig. 10) is above its upper bound $\overline{s}_i^{\mathrm{g}}$ (red bold curve in Fig. 10), AMV $i+1$ will collide with AMV i inevitably even if AMV i uses the maximum acceleration rate $\overline{a}$ to accelerate while AMV $i+1$ using the minimum acceleration rate $\underline{a}$ to decelerate. Constraints (3) is violated and thus the DP-TPP is not feasible. On the other hand, Theorem 3 proves that when the initial boundary conditions are admissible feasible solutions satisfying constraints (1) - (5) exist. This proves the sufficiency.

Theorem 5. *If the DP-TPP is feasible, we can find a feasible solution with the PH algorithm when* $T=t^{\mathrm{d}}$.

Proof: The PH algorithm seeks the minimum feasible acceleration rate for each AMV using a binary search within $\left[\underline{a},\overline{a}\right]$. If the current acceleration rate is too small to be feasible, a larger one will be used instead. And if the current acceleration rate is large and feasible, a smaller one will be tested until the minimum feasible rate is found to obtain smoother trajectories. The first AMV's trajectory $X_1^{\mathrm{P}\cdot t^{\mathrm{d}}}$ will follow its global feasible cone lower bound. And the following AMVs' trajectory $\{X_i^{\mathrm{P}\cdot t^{\mathrm{d}}}\}_{i\in\mathscr{I}\setminus\{1\}}$ will accelerate and merge into the preceding AMVs' shadow trajectories with the minimum feasible acceleration rates. Similar to the proof of Theorem 3, we can easily show that $X_1^{\mathrm{P}\cdot t^{\mathrm{d}}}$ and $\{X_i^{\mathrm{P}\cdot t^{\mathrm{d}}}\}_{i\in\mathscr{I}\setminus\{1\}}$ obtained from the PH algorithm are also feasible to constraints (1) - (5) when $T=t^{\mathrm{d}}$. This completes the proof.

After the above feasibility analysis, we explore the optimality properties of the DP-TPP by constructing heuristic alternative solutions $\widetilde{x}_i,\forall i\in\mathscr{I}$, to solve a special case of the DP-TPP to the exact optimum when $c=0$. With t^{d} derived in Equation (11), the SP with $c=0$ is reformulated as follows.

$$\mathrm{SP}:\min_{a_i(t)}\frac{1}{2}\sum_{i=1}^{I}\int_0^{t^{\mathrm{d}}}a_i^{\,2}(t)dt,$$

subject to constraints (1)-(5), where $a_i(t)=\dot{v}_i(t)$ denotes the acceleration (or deceleration) rate of AMV i at time t, and $v_i(t)=\dot{x}_i(t)$ denotes the speed of AMV i at time t. Consider the SP for only one AMV $i,\forall i\in\mathscr{I}$, safety constraints (3), are relaxed. Thus, the relaxed SP (RSP) is decomposed as follows.

$$\mathrm{RSP}:\ \min_{a_i(t)}\int_0^{t^{\mathrm{d}}}a_i^{\,2}(t)dt,\forall i\in\mathscr{I}$$

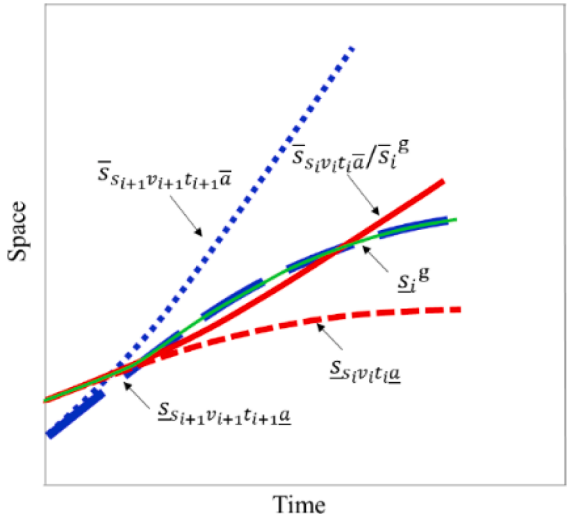


Fig. 10. An illustration of inadmissible initial boundary conditions.

subject to constraints (1), (2), (4), and (5).

With $\{a_i\}$ as control inputs and $\{v_i, x_i\}$ as state variables, the Hamiltonian equation is given as follows.

$$H_i(t, x_i(t), v_i(t), a_i(t)) = {a_i}^2(t) + \lambda_{i,1}(t)v_i(t) + \lambda_{i,2}(t)a_i(t).$$

The Pontryagin's maximum principle costate equation yields Equations (18)-(19), and the necessary condition for optimality yields Equation (20).

$$\frac{\partial H_i(t)}{\partial x_i(t)} + \dot{\lambda}_{i,1}(t) = 0 + \dot{\lambda}_{i,1}(t) = 0. \tag{18}$$

$$\frac{\partial H_i(t)}{\partial v_i(t)} + \dot{\lambda}_{i,2}(t) = \lambda_{i,1}(t) + \dot{\lambda}_{i,2}(t) = 0. \tag{19}$$

$$\frac{\partial H_i(t)}{\partial a_i(t)} = 2a_i(t) + \lambda_{i,2}(t) = 0 \tag{20}$$

Equations (18)-(20) yield $a_i(t) = -\frac{\lambda_{i,2}(t)}{2} = m_i t + n_i$. When acceleration and speed constraints are not violated, AMV i's acceleration is a linear function of time t. Speed is a quadratic function of time t. Location is a cubic function of time t.

$$\widetilde{a}_i(t) = m_i t + n_i.$$

$$\widetilde{v}_i(t) = \frac{1}{2}m_i t^2 + n_i t + k_i,$$

$$\widetilde{x}_i(t) = \frac{1}{6}m_i t^3 + \frac{1}{2}n_i t^2 + k_i t + q_i,$$

where $m_i = \frac{12\times x_i(t^d) - 12\times v_i^- \times t^d - 12\times s_i^- + 6\times t^d \times v_i^- - 6\times t^d \times v^d}{-{t^d}^3}$, $n_i = \frac{2\times v^d - 2\times v_i^- - m_i \times {t^d}^2}{2\times t^d}$, $k_i = v_i^-$ and $q_i = s_i^-$.

If acceleration constraints are violated, the segment(s) with the acceleration equal to the boundary value(s) must be added at the start and/or the end of the acceleration function.

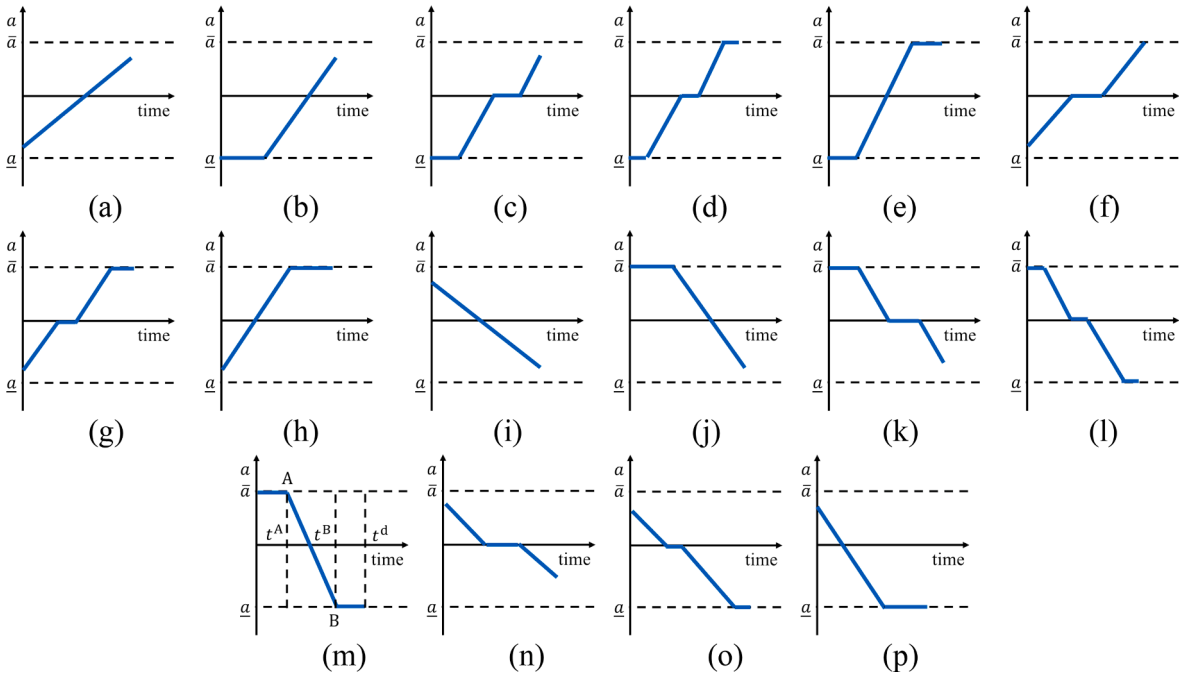


Fig. 11. Heuristic alternative acceleration solutions.

$$\widetilde{a}_i(t) = \begin{cases} \overline{a}, if -\dfrac{\lambda_{i,2}(t)}{2} > \overline{a}; \\ \underline{a}, if -\dfrac{\lambda_{i,2}(t)}{2} < \underline{a}; \\ m_i t + n_i, if \underline{a} < -\dfrac{\lambda_{i,2}(t)}{2} < \overline{a}. \end{cases}$$

It is noted that the jerk is always equal to 0 or a constant value $\frac{-\dot{\lambda}_{i,2}(t)}{2} = \frac{\lambda_{i,1}(t)}{2}$. If speed constraints are violated, Equation (19) does not hold. The segment with zero acceleration must be added in the middle of the acceleration function, where the speed boundary value is reached. Specifically, if the slope of the original acceleration (or jerk) $-\frac{\dot{\lambda}_{i,2}(t)}{2}$ is greater than 0, $\lambda_{i,1}(t)$ must be greater than 0 so that Equation (19) holds. Adding the segment with the acceleration equal to 0 in the middle leads to $\lambda_{i,1}(t) + \dot{\lambda}_{i,2}(t) \rangle 0$. In this case, AMV i's speed shall be revised to $\underline{v}$. If the original jerk is less than 0, $\lambda_{i,1}(t)$ must be less than 0 so that Equation (19) holds. Adding the segment with the acceleration equal to 0 in the middle would lead to $\lambda_{i,1}(t) + \dot{\lambda}_{i,2}(t) \langle 0$. In this case, AMV i's speed shall be revised to $\overline{v}$.

$$\widetilde{v}_i(t) = \begin{cases} \underline{v}, if \lambda_{i,1}(t) + \dot{\lambda}_{i,2}(t) \rangle 0; \\ \overline{v}, if \lambda_{i,1}(t) + \dot{\lambda}_{i,2}(t) \langle 0; \\ \dfrac{1}{2} m_i t^2 + n_i t + v^c, if \lambda_{i,1}(t) + \dot{\lambda}_{i,2}(t) = 0. \end{cases}$$

There are 16 possible heuristic alternative solutions for each AMV, i.e., 16 acceleration functions, 16speed functions, and 16 location functions. Given the space limit, only the acceleration functions are plotted for illustration.

Fig. 11 (m) shows that two horizontal segments are added at the beginning and end of the acceleration function. The two critical points are marked as A and B. The speed of A v_i^{A} and location of A x_i^{A} are calculated as follows.

$$v_i^{\mathrm{A}} = v_i(0) + \overline{a} \times t^{\mathrm{A}}$$

$$x_i^{\mathrm{A}} = x_i(0) + v^c \times t^{\mathrm{A}} + \frac{1}{2} \times \overline{a} \times \left(t^{\mathrm{A}}\right)^2$$

The PMP analysis shows that the acceleration, speed, and location functions between A and B are linear, quadratic, and cubic, respectively.

$$a_i(t) = m^{'} \times \left(t - t^{\mathrm{A}}\right) + n^{'}$$

$$v_i(t) = \frac{1}{2} \times m^{'} \times \left(t - t^{\mathrm{A}}\right)^2 + n^{'} \times \left(t - t^{\mathrm{A}}\right) + v^{\mathrm{A}}$$

$$x_i(t) = \frac{1}{6} \times m^{'} \times \left(t - t^{\mathrm{A}}\right)^3 + \frac{1}{2} \times n^{'} \times \left(t - t^{\mathrm{A}}\right)^2 + v^{\mathrm{A}} \times \left(t - t^{\mathrm{A}}\right) + x^{\mathrm{A}}$$

where $n = \overline{a}$ and $m = \frac{\overline{a} - \underline{a}}{t^{\mathrm{A}} - t^{\mathrm{B}}}$.

The speed of B v_i^{B} and location of B x_i^{B} are calculated as follows.

$$v_i^{\mathrm{B}} = \frac{1}{2} \times m^{'} \times \left(t^{\mathrm{B}} - t^{\mathrm{A}}\right)^2 + n^{'} \times \left(t^{\mathrm{B}} - t^{\mathrm{A}}\right) + v^{\mathrm{A}}$$

$$x_i^{\mathrm{B}} = \frac{1}{6} \times m^{'} \times \left(t^{\mathrm{B}} - t^{\mathrm{A}}\right)^3 + \frac{1}{2} \times n^{'} \times \left(t^{\mathrm{B}} - t^{\mathrm{A}}\right)^2 + v^{\mathrm{A}} \times \left(t^{B} - t^{\mathrm{A}}\right) + x^{\mathrm{A}}$$

The speed $v_i\left(t^{\mathrm{d}}\right)$ and location $x_i\left(t^{\mathrm{d}}\right)$ by the end of the operation are calculated as follows.

$$v_i\left(t^{\mathrm{d}}\right) = v^{\mathrm{B}} + \underline{a} \times \left(t^{\mathrm{d}} - t^{B}\right)$$

$$x_i\left(t^{\mathrm{d}}\right) = x^{\mathrm{B}} + v^{\mathrm{B}} \times \left(t^{\mathrm{d}} - t^{\mathrm{B}}\right) + \frac{1}{2} \times \underline{a} \times \left(t^{\mathrm{d}} - t^{\mathrm{B}}\right)^2$$

With the above two equations, t^{A} and t^{B} can be easily computed. For other cases in Fig. 11, the computation follows similar logic and thus is omitted here.

Although safety constraints are not explicitly included in the RSP, the above-proposed heuristic alternative solution can be feasible when the initial boundary conditions are properly set so that the consecutive vehicles do not collide (Malikopoulos et al., 2018). Yet, the PH algorithm is still needed for solving general DP-TPP instances to near-optimal trajectories in real-time. To test the performance of the PH algorithm, Section 5 discretizes the DP-TPP into a quadratic programming model to obtain the exact optimal solution for comparison.

5. Discrete modeling and analysis

With the minimum docking time t^{d} obtained above (which is the optimal solution to the FP), we discretize the DP-TPP (or the SP by fixing $T = t^{d}$) into a quadratic programming model (QPM) to obtain the exact optimal solution to the SP and compare the solution with the PH solution to test the PH performance. The feasible region of the QPM is greatly reduced after adding valid cuts derived from the global feasible cone.

We use unit time interval δ (e.g., 0.1 s) to discretize the time horizon $[0, t^{d} + \delta]$ into $J+2$ points indexed by $\mathscr{J} = [0, 1, 2, \cdots, J+1]$ where $J = \lceil t^{d}/\delta \rceil$ such that an integer time j represents the actual time point $j\hat{A}\cdot\delta$. Let decision variables $X = \{x_{ij}\}_{\forall i\in\mathscr{I}, j\in\mathscr{J}}$ denote the position of AMVs at all time points. With this, the QPM is formulated as:

$$\text{QPM}: \min_{x_{ij}\in X} \sum_{i=1}^{I}\sum_{j=0}^{J} \delta \times \left(\frac{\frac{x_{i(j+2)}-x_{i(j+1)}}{\delta} - \frac{x_{i(j+1)}-x_{ij}}{\delta}}{\delta} \right)^{2} + c \times \sum_{i=1}^{I} \sum\nolimits_{j=1}^{J} \delta \times \left(j \times \delta \times \overline{v} - \left(x_{ij} - x_{i0} \right) \right),$$

subject to the following constraints (21) - (25) corresponding to constraints (1) - (5):

$$0 \le \frac{x_{i(j+1)} - x_{ij}}{\delta} \le \overline{v} \forall i \in \mathscr{I}, j \in \mathscr{J} \backslash \{J+1\}. \tag{21}$$

$$\underline{a} \le \frac{\frac{x_{i(j+2)}-x_{i(j+1)}}{\delta} - \frac{x_{i(j+1)}-x_{ij}}{\delta}}{\delta} \le \overline{a}, \forall i \in \mathscr{I}, j \in \mathscr{J} \backslash \{J, J+1\}. \tag{22}$$

$$x_{(i-1)j} - x_{ij} \ge l \forall i \in \mathscr{I} \backslash \{1\}, j \in \mathscr{J}. \tag{23}$$

$$x_{i0} = s_i^{-}, \frac{x_{i1} - x_{i0}}{\delta} = v_i^{-}, \ \forall i \in \mathscr{I},$$

$$\frac{x_{i(J+1)} - x_{iJ}}{\delta} = v^{d}, \ x_{(i-1)J} - x_{iJ} = l \forall i, \mathscr{I} \backslash \{1\}.$$

The global feasible cone yields another constraint set as the following valid cuts:

$$\underline{s}_{ij}^{g} \le x_{ij} \le \overline{s}_{ij}^{g}, \forall i \in \mathscr{I}, j \in \mathscr{J}. \tag{26}$$

$\underline{s}_{ij}^{g}$ is equal to $\underline{s}_{i}^{g}(t)$ where $t = j\hat{A}\cdot\delta$ and denotes the global feasible cone lower bound of AMV i. Similarly, $\overline{s}_{ij}^{g}$ is the global feasible cone upper bound. Note that as δ goes to an infinitesimal value, the QPM solution converges to the optimal DP-TPP solution.

Theorem 6. *The objective function of the QPM is convex.*

Proof: Let $F(X)$ denote the objective function of the QPM. Define $W_1 := \{x_{ij}^{1}\}_{\forall i\in\mathscr{I}, j\in\mathscr{J}}$, $W_2 := \{x_{ij}^{2}\}_{\forall i\in\mathscr{I}, j\in\mathscr{J}}$ and $W_3 := \left[\left(\{x_{ij}^{1}\}_{\forall i\in\mathscr{I}, j\in\mathscr{J}} + \{x_{ij}^{2}\}_{\forall i\in\mathscr{I}, j\in\mathscr{J}} \right) \Big/ 2 \right]$. Proving function $F(X)$ is convex is equivalent to proving $0.5F(W_1) + 0.5F(W_2) \ge F(W_3)$.

$$\begin{aligned}
&0.5F(W_1) + 0.5F(W_2) - F(W_3) \\
&= 0.5\left[\sum_{i=1}^{I}\sum_{j=0}^{J} \delta \times \left(\frac{x_{i(j+1)}^{1} - x_{ij}^{1}}{\delta} \right)^{2} + c \times \sum_{i=1}^{I} \sum\nolimits_{j=1}^{J} \delta \times \left(j \times \delta \times \overline{v} - \left(x_{ij}^{1} - x_{i0}^{1} \right) \right) \right] + 0.5\left[\sum_{i=1}^{I}\sum_{j=0}^{J} \delta \times \left(\frac{x_{i(j+1)}^{2} - x_{ij}^{2}}{\delta} \right)^{2} + c \right. \\
&\left. \times \sum_{i=1}^{I} \sum\nolimits_{j=1}^{J} \delta \times \left(j \times \delta \times \overline{v} - \left(x_{ij}^{2} - x_{i0}^{2} \right) \right) \right] - \sum_{i=1}^{I}\sum_{j=0}^{J} \delta \times \left(\frac{x_{i(j+1)}^{1} + x_{i(j+1)}^{2}}{2*\delta} - \frac{x_{ij}^{1} + x_{ij}^{2}}{2*\delta} \right)^{2} - c \times \sum_{i=1}^{I} \sum\nolimits_{j=1}^{J} \delta \times \left(j \times \delta \times \overline{v} \right. \\
&\left. - \left(\frac{x_{ij}^{1} + x_{ij}^{2}}{2} - \frac{x_{i0}^{1} + x_{i0}^{2}}{2} \right) \right) \\
&= 0.25\left[\sum_{i=1}^{I}\sum_{j=0}^{J} \delta \times \left(\frac{x_{i(j+1)}^{1} - x_{ij}^{1}}{\delta} \right)^{2} + \sum_{i=1}^{I}\sum_{j=0}^{J} \delta \times \left(\frac{x_{i(j+1)}^{2} - x_{ij}^{2}}{\delta} \right)^{2} - 2\sum_{i=1}^{I}\sum_{j=0}^{J} \delta \times \left(\frac{x_{i(j+1)}^{1} - x_{ij}^{1}}{\delta} \right) \times \left(\frac{x_{i(j+1)}^{2} - x_{ij}^{2}}{\delta} \right) \right] \\
&= 0.25\sum_{i=1}^{I}\sum_{j=0}^{J} \delta \times \left[\left(\frac{x_{i(j+1)}^{1} - x_{ij}^{1}}{\delta} \right) - \left(\frac{x_{i(j+1)}^{2} - x_{ij}^{2}}{\delta} \right) \right]^{2} \\
&\ge 0
\end{aligned}$$

This completes the proof.

The convexity of the objective function ensures that the exact optimal solution can be obtained. Since the objective is just a second-order function of the decision variables, this model is a convex quadratic programming model and can be solved to the exact optimum by commercial solvers.

6. Numerical experiments

This section first tests the performance of the near-optimal PH algorithm with that of the exact QPM approach. The QPM instances are solved by a commercial optimization solver, Gurobi. Further, numerical experiments compare the PH algorithm results with the CACC model results to test the superiority of optimizing the AMV docking trajectories with the proposed approach. Finally, sensitivity analysis is conducted on key parameters to draw some insights. These numerical examples are conducted on a PC with Windows 10 operating system with a 2.6 GHz CPU and 16 GB RAM. The code is programmed in MATLAB.

We test the approaches on a number of instances with different parameter settings. In the default instance, we set $= 10, \bar{v} = 30\text{m/s}$, $v^{\text{d}} = 28\text{m/s}$, $\bar{a} = 2\text{m/s}^2$, $\underline{a} = -2\text{m/s}^2$, $l_i = 4\text{m}$ and $\delta = 0.1\text{s}$. Further, set $c = 0.1$ to have the accumulative uncovered travel distance and squared acceleration cost components comparable, and making different cost components comparable in an objective with multiple components is a standard modeling treatment in the literature (Wei et al., 2016). We also randomly generate initial speeds with an average $v_{avg} = 24\text{m/s}$ and initial locations with an average car-following gap $gap_{avg} = 12\text{m}$, and select the ones that are admissible as the initial boundary conditions.

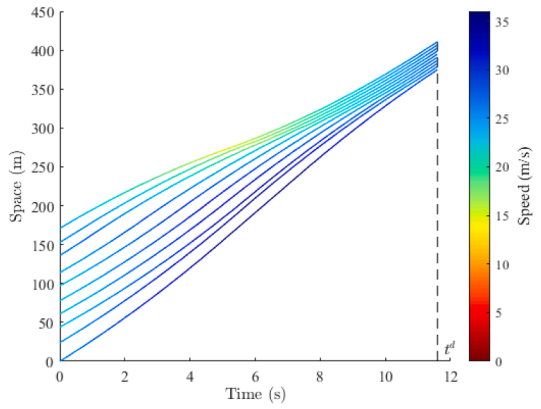
6.1. Comparison with the QPM

The optimality comparison results of the PH and the QPM are shown in Table 1, where F_{QPM} is the SP exact optimal objective from the QPM solution, $f_{\text{QPM}_{\text{UTD}}}$ the accumulative uncovered travel distance component of the SP exact optimal objective from the QPM solution, $f_{\text{QPM}_{\text{SA}}}$ the squared acceleration component of the SP exact optimal objective value from the QPM solution, F_{PH} the SP near-optimal objective from the PH algorithm, $f_{\text{PH}_{\text{UTD}}}$ the accumulative uncovered travel distance component of the SP near-optimal objective from the PH algorithm, $f_{\text{PH}_{\text{SA}}}$ the squared acceleration component of the SP near-optimal objective from the PH algorithm, $\mathscr{E} = \frac{F_{\text{PH}} - F_{\text{QPM}}}{F_{\text{QPM}}} \times 100\%$ the SP optimality gap between these two objectives, $\mathscr{E}_{\text{UTD}} = \frac{f_{\text{PH\_UTD}} - f_{\text{QPM\_UTD}}}{f_{\text{QPM\_UTD}}} \times 100\%$ the optimality gap between these two uncovered travel distance components, and $\mathscr{E}_{\text{SA}} = \frac{f_{\text{PH\_AS}} - f_{\text{QPM\_AS}}}{f_{\text{QPM\_AS}}} \times 100\%$ the optimality gap between these two squared acceleration components. From Table 1, the $\mathscr{E}$ values indicate that the SP objective of the PH is only 5.2% worse than that of the QPM on average, and the performance of the PH is relatively robust, with $\mathscr{E}$ values less than 7% across all instances. The $\mathscr{E}_{\text{UTD}}$ values are even smaller, with a mean of 3.3%, indicating that the PH's mobility component solution is very close to the true optimum. The $\mathscr{E}_{\text{SA}}$ values are relatively high, with an average of 9.8%. Note that some $\mathscr{E}_{\text{SA}}$ values are negative, which indicates that the PH solution may have better riding comfort performance than the QPM solution in certain instances. Yet the maximum $\mathscr{E}_{\text{SA}}$ value is close to 22% when $\bar{v} = 36$ m/s. This is because the sum of the squared acceleration term tends to be dominated by trajectory segments with relatively high acceleration values. Thus its value is sensitive to acceleration differences in a small portion of trajectory segments. To illustrate this point, the trajectories from the PH and the QPM when $\bar{v} = 36$ m/s are plotted in Fig. 12. The two sets of trajectories have similar shapes across most segments. The last few AMVs from the PH merge into preceding AMVs' shadow trajectories earlier than the QPM. Further, we want to point out that this squared acceleration is relatively subjective (or not exactly accurate) in reflecting riding comfort or energy consumption, and it comprises a relatively small portion of the total objective value. Thus, its difference from the true optimum is relatively acceptable. Overall, these results suggest that the PH solution is near-optimal and acceptable for engineering practice.

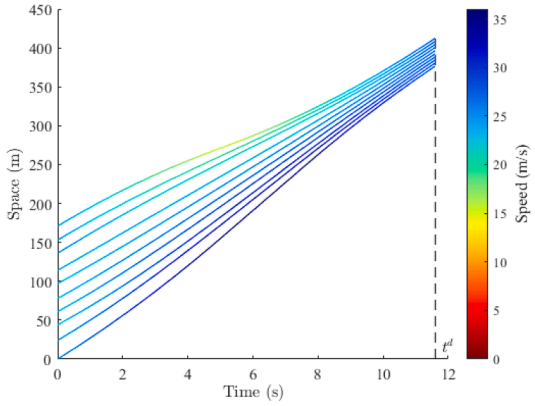
After comparing the solution optimality, Table 2 shows the solution times for these different approaches. Time_{QPM} is the QPM

Table 1
The PH and the QPM optimality comparison.

Parameter value	t^{d}(s)	F_{QPM}	$f_{\text{QPM\_UTD}}$	$f_{\text{QPM\_SA}}$	F_{PH}	$f_{\text{PH\_UTD}}$	$f_{\text{PH\_SA}}$	$\mathscr{E}$ (%)	$\mathscr{E}_{\text{UTD}}$(%)	$\mathscr{E}_{\text{SA}}$ (%)
$I = 5$ veh	9.4	154	94	60	162	97	65	5.2	3.2	8.3
$I = 10$ veh (default)	14.1	876	708	168	918	722	196	4.8	2.0	16.7
I = 15 veh	19.4	2173	1831	342	2287	1937	350	5.2	5.8	2.3
I = 30 veh	27.9	11,398	9944	1454	12,083	10,942	1141	6.0	10.0	−21.5
$v_{avg} = 18$ m/s	15.2	1290	1074	216	1351	1105	246	4.7	2.9	13.9
$v_{avg} = 22$ m/s	14.5	929	751	178	974	770	204	4.8	2.5	14.6
$v_{avg} = 26$ m/s	13.7	808	619	189	849	626	223	5.1	1.1	18.0
$gap_{avg} = 4\ m$	10.6	434	306	128	460	310	150	6.0	1.3	17.2
$gap_{avg} = 20\ m$	17.8	1383	1151	232	1460	1196	264	5.6	3.9	13.8
$gap_{avg} = 36\ m$	24.5	2670	2311	359	2809	2498	311	5.2	8.1	−13.4
$v^{\text{d}} = 20\text{m/s}$	13.5	828	674	154	853	677	176	3.0	0.4	14.3
$v^{\text{d}} = 26\text{m/s}$	14.4	781	619	162	819	625	194	4.9	1.0	19.8
$v^{\text{d}} = 30\text{m/s}$	15.1	907	717	190	960	730	230	5.8	1.8	21.1
$\bar{v} = 28\text{m/s}$	16.0	833	646	187	888	670	218	6.6	3.7	16.6
$\bar{v} = 32\text{m/s}$	13.3	854	693	161	901	707	194	5.5	2.0	20.5
$\bar{v} = 36\text{m/s}$	11.6	840	674	166	880	677	203	4.8	0.4	22.3
$\bar{a} = 1\text{m/s}^2$	18.4	957	875	82	1006	937	69	5.1	7.1	−15.9
$\bar{a} = 1.5\text{m/s}^2$	15.3	905	787	118	934	817	117	3.2	3.8	−0.8
$\bar{a} = 2.5\text{m/s}^2$	13.5	757	554	203	805	564	241	6.3	1.8	18.7
Average	15.7							5.2	3.3	9.8



(a) Trajectories from the PH when $\bar{v} = 36$ m/s.



(b) Trajectories from the QPM when $\bar{v} = 36$ m/s.

Fig. 12. Trajectories from the PH and the QPM.

Table 2
The PH and the QPM solution time comparison.

Parameter value	t^d(s)	Time_{QPM}(s)	$\text{Time}_{\text{QPM\_Cone}}$(s)	Time_{PH}(s)
$I = 5$ veh	9.4	0.180	0.144	0.003
$I = 10$ veh (default)	14.1	0.351	0.193	0.005
I = 15 veh	19.4	0.714	0.335	0.008
I = 30 veh	27.9	3.249	0.856	0.022
$v_{avg} = 18$ m/s	15.2	0.317	0.204	0.006
$v_{avg} = 22$ m/s	14.5	0.327	0.195	0.006
$v_{avg} = 26$ m/s	13.7	0.328	0.198	0.006
$gap_{avg} = 4\ m$	10.6	0.252	0.181	0.005
$gap_{avg} = 20\ m$	17.8	0.428	0.227	0.006
$gap_{avg} = 36\ m$	24.5	0.608	0.277	0.005
$v^d = 20$m/s	13.5	0.237	0.175	0.006
$v^d = 26$m/s	14.4	0.311	0.198	0.006
$v^d = 30$m/s	15.1	0.319	0.199	0.005
$\bar{v} = 28$m/s	16.0	0.351	0.193	0.005
$\bar{v} = 32$m/s	13.3	0.287	0.187	0.006
$\bar{v} = 36$m/s	11.6	0.275	0.181	0.006
$\bar{a} = 1$m/s^2	18.4	0.479	0.245	0.008
$\bar{a} = 1.5$m/s^2	15.3	0.519	0.257	0.006
$\bar{a} = 2.5$m/s^2	13.5	0.340	0.193	0.006
Average	15.7	0.519	0.244	0.007

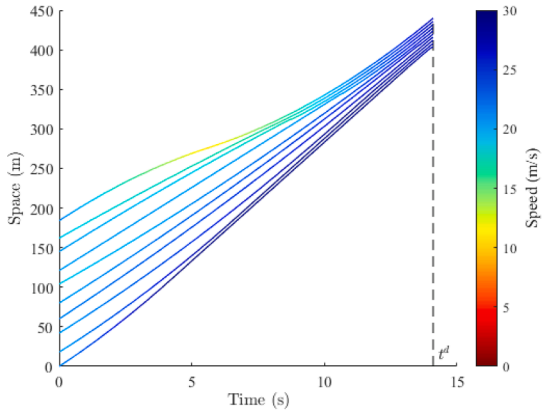
Table 3
The PH and the CACC optimality comparison.

Parameter value	t^{d}_{CACC}(s)	F_{CACC}	f_{CACC_UTD}	f_{CACC_SA}	t^{d}_{PH}(s)	F_{PH}	f_{PH_UTD}	f_{PH_SA}	$\mathscr{E}_{time}$(%)	$\mathscr{E}$ (%)	$\mathscr{E}_{UTD}$(%)	$\mathscr{E}_{SA}$(%)
$I = 10$ veh (default)	20.3	2107	1162	945	14.1	876	708	168	44.0	140.5	64.1	462.5
$I = 30$ veh	70.1	51,269	46,878	4391	27.9	11,398	9944	1454	151.3	349.8	371.4	202.0
$v_{avg} = 18$ m/s	25.5	3221	2459	762	15.2	1290	1074	216	67.8	149.7	129.0	252.8
$v_{avg} = 26$ m/s	24.5	2218	1488	730	13.7	808	619	189	78.8	174.5	140.4	286.2
$gap_{avg} = 4\ m$	20.1	1763	1240	523	10.6	434	306	128	89.6	306.2	305.2	308.6
$gap_{avg} = 20\ m$	28.2	3669	2701	968	17.8	1383	1151	232	58.4	165.3	134.7	317.2
$v^{d} = 20$m/s	22.3	3469	2404	1065	13.5	828	674	154	65.2	319.0	256.7	591.6
$v^{d} = 26$m/s	23	2808	1956	852	14.4	781	619	162	59.7	259.5	216.0	425.9
$\bar{v} = 32$m/s	21.6	2599	1865	734	13.3	854	693	161	62.4	204.3	169.1	355.9
$\bar{v} = 36$m/s	21.7	3353	2760	593	11.6	840	674	166	87.1	299.2	309.5	257.2
$\bar{a} = 1$m/s^2	27.9	2556	2292	264	18.4	957	875	82	51.6	167.1	161.9	222.0
$\bar{a} = 1.5$m/s^2	21.7	1992	1559	433	15.3	905	787	118	41.8	120.1	98.1	266.9
Average	27.2				15.5				71.5	221.3	196.3	329.1

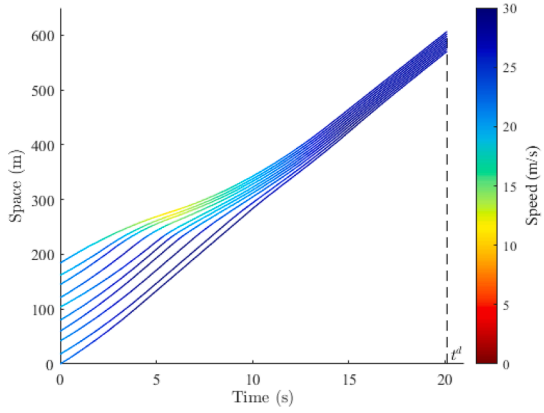
solution time without the global feasible cone constraints (26), $\text{Time}_{\text{QPM\_Cone}}$ the QPM solution time with the global feasible cone constraints (26), and Time_{PH} the solution time of the PH algorithm. We see that, on average, the PH solution time is only 0.007 s, and the QPM solution times are 0.519 s and 0.244 s without and with cutting the feasible region, respectively. First, it is noted that the solution time of the QPM can be greatly reduced by around 53% after introducing the global feasible cone cuts. This demonstrates the power of using analytical properties to expedite the solution efficiency of a numerical solution approach, which is helpful for applications that require absolute optimality or just to solve the true optimum as a benchmark for comparison. Further, if real-time applications require even greater solution efficiency, the results indicate that the PH is a suitable approach. As AMV fleet size I increases, the solution time of the QPM increases significantly. However, the solution time of the PH is kept within 0.1 s. For the other instances, the solution time of the QPM ranges from 0.175 s to 0.277 s with cutting the feasible region, yet the solution time of the PH can still be within 0.008 s. Overall, with slight sub-optimality yet millisecond-level solution efficiency, the PH is an appealing approach for real-time AMV applications.

6.2. Comparison with the CACC

To show the advantage of the proposed heuristic approach in terms of planning vehicle trajectories over the existing benchmark, we adapt the CACC model proposed by Milanés and Shladover (2014) to solve a basic docking problem without the FP optimization, i.e., the minimum docking time limitation. In the benchmark case, we let the first vehicle follow the same trajectory generated by the PH algorithm and populate the following trajectories with the CACC model. The comparison results between the benchmark and the PH solution are presented in Table 3 and Fig. 13. $t^{\text{d}}_{\text{CACC}}$ is the docking time needed in the benchmark solution, t^{d}_{PH} the docking time needed in the PH solution, $\mathscr{E}_{\text{time}} = \frac{t^{\text{d}}_{\text{CACC}} - t^{\text{d}}_{\text{PH}}}{t^{\text{d}}_{\text{PH}}} \times 100\%$ the FP optimality gap between these two objectives, F_{CACC} the SP objective value in the benchmark solution, $f_{\text{CACC\_UTD}}$ the uncovered travel distance component in the benchmark SP objective, $f_{\text{CACC\_SA}}$ the squared acceleration component in the benchmark SP objective, $\mathscr{E} = \frac{F_{\text{CACC}} - F_{\text{PH}}}{F_{\text{PH}}} \times 100\%$ the SP optimality gap between these two objectives,



(a) Trajectories from the PH when $\mathcal{I}$ =10.



(b) Trajectories from the CACC when $\mathcal{I}$ =10.

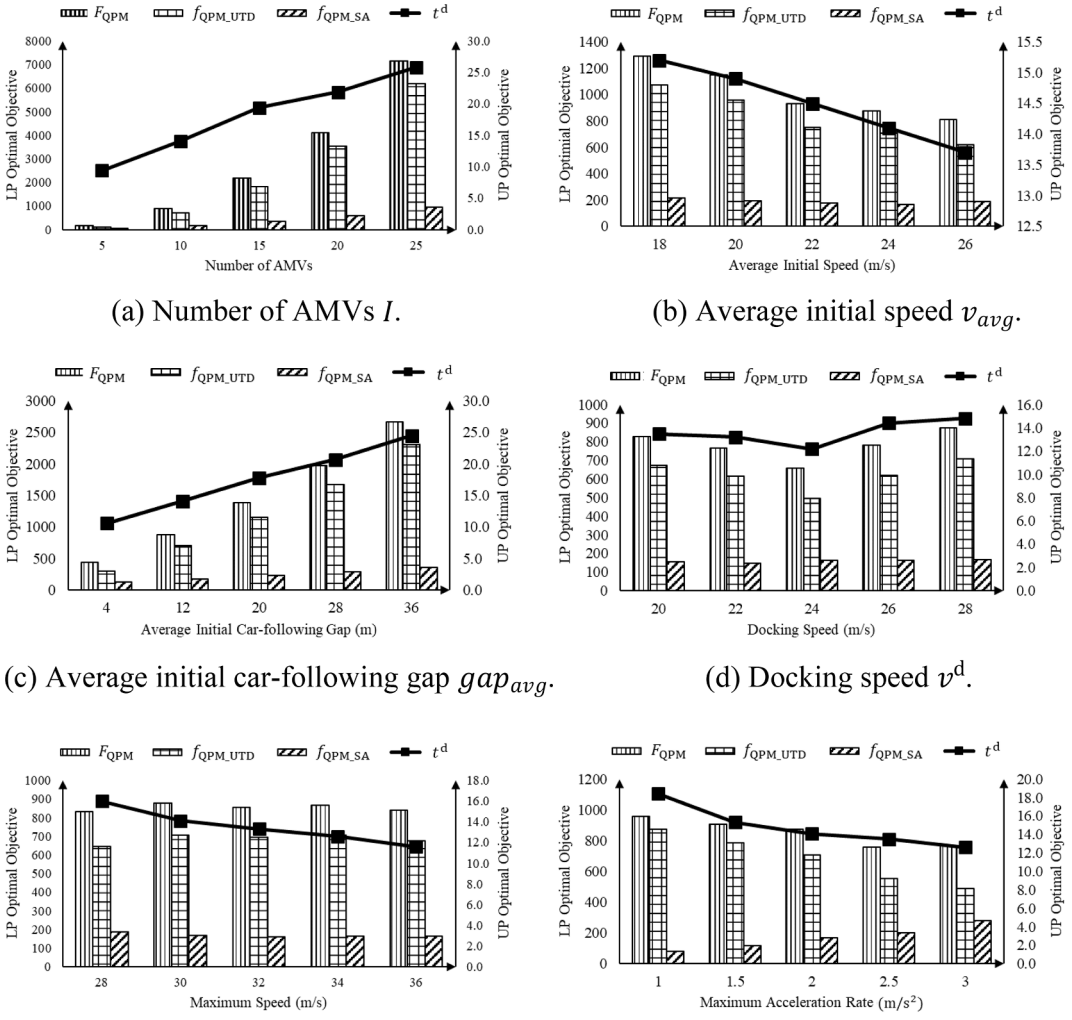
Fig. 13. Trajectories from the PH and the CACC.

$\mathscr{E}_{\text{UTD}} = \frac{f_{\text{CACC\_UTD}} - f_{\text{PH\_UTD}}}{f_{\text{PH\_UTD}}} \times 100\%$ the optimality gap between these two accumulative uncovered travel distance components, and $\mathscr{E}_{\text{SA}} = \frac{f_{\text{CACC\_AS}} - f_{\text{PH\_AS}}}{f_{\text{PH\_AS}}} \times 100\%$ the optimality gap between these two squared acceleration components.

As shown in Table 3, the FP and SP optimality gaps between the CACC and the PH are significant, with an average of 71.5% and 221.3%, respectively. Across all instances with different settings, the UTD optimality gap is significant with an average of 196.3%, and the SA optimality gap is even greater with an average of 329.1%. This clearly shows that the proposed trajectory optimization approach yields better performance in docking (or platooning) vehicles than the existing benchmark. Further, we see that at greater I, v_{avg}, $\bar{v}$ and l values and at a lower gap_{avg}, v^{d} and $\bar{a}$ values optimality gap $\mathscr{E}$ is greater. This indicates that the advantage of the proposed method will be further articulated when the fleet (or platoon) size is large, the initial speeds are high (which is likely on the freeway), the speed limit is high (which is also likely on the freeway), the initial car-following gaps are small (when AMVs are close to each other), the vehicle power is low (so is the acceleration limit), and the docking (or platooning) speed is low (which is likely in the early stage of the technology for safety reasons). As illustrated in Fig. 13 (b), the benchmark trajectories have substantially longer tails adjusting AMV speeds to dock (or platoon) them despite considerable speed adjustment at the early stage. In contrast, the proposed approach adjusts trajectory speeds at a relatively homogeneous pace throughout the docking operation. This explains why the proposed approach is superior in docking time and riding comfort.

6.3. Sensitivity analysis

This subsection conducts sensitivity analysis on key parameters. A number of instances are constructed, each varying only one parameter value and keeping the others the same as the default case. Fig. 14 plots how the optimal objective and its two components



(a) Number of AMVs I.

(b) Average initial speed v_{avg}.

(c) Average initial car-following gap gap_{avg}.

(d) Docking speed v^{d}.

(e) Maximum speed $\bar{v}$.

(f) Maximum acceleration $\bar{a}$.

Fig. 14. Optimal objectives evolving trends.

vary as each parameter changes. From Fig. 14 (a), we notice that the optimal objective F_{QPM} and its components $f_{\text{QPM\_UTD}}$ and $f_{\text{QPM\_SA}}$ all increase super-linearly as the fleet size I increases while $f_{\text{QPM\_SA}}$ increases at a milder pace. The super-linear growth indicates that the cost and complexity for each AMV in the docking operation increase as the AMV fleet size grows, which implies the need for rigorous optimization for docking a relatively large AMV fleet. Also, the increasing t^{d} suggests that a bigger AMV fleet can be separated into several sub-fleets with a smaller number of AMVs. Thus, the proposed docking operation can be implemented into each sub-fleet, and then each sub-fleet can be viewed as a longer AMV that needs to be docked with other sub-fleets to reduce the docking time as well as cost. Fig. 14 (b) shows that as the average initial speed v_{avg} increases, F_{QPM} and $f_{\text{QPM\_UTD}}$ decrease and t^{d} is also decreasing slightly. This is intuitive since high initial speeds indicate less speed difference from the docking speed and enable the following AMVs to catch up with their preceding AMVs quickly with less loss of mobility. $f_{\text{QPM\_SA}}$ is relatively stable, suggesting that riding comfort can be well guaranteed under different initial speed conditions. In Fig. 14 (c), t^{d}, F_{QPM}, $f_{\text{QPM\_UTD}}$ and $f_{\text{QPM\_SA}}$ increase with the average initial car-following gap gap_{avg}, as shown. It is intuitive that an increase in the initial gap gap_{avg} means a longer distance to catch up with so that it will lead to a longer docking time and traveled distance and thus $f_{\text{QPM\_UTD}}$ as well. As the initial car-following gap gets larger, the first few AMVs need to adjust their speed more (even stop for a while when it is necessary) to wait for the following AMVs, which leads to a larger $f_{\text{QPM\_SA}}$. Fig. 14 (d) shows that when the docking speed is close to the average initial speed (i.e., 24 m/s), the cost of the docking operation is the minimum for both the FP and the SP shown as the dents. As the difference between the docking speed and the average initial speed increases, t^{d}, F_{QPM} and $f_{\text{QPM\_UTD}}$ increase. Because more speed adjustments are needed when the speed difference is larger, thus leading to a longer docking time and more loss of mobility. Riding comfort can still be kept relatively constant even with large speed differences. To achieve greater docking efficiency and lower cost, choosing a docking speed close to AMVs' average initial speed is better. In Fig. 14 (e), as the maximum speed $\bar{v}$ increases, mobility and riding comfort, indicated by $f_{\text{QPM\_UTD}}$ and $f_{\text{QPM\_SA}}$ remain relatively constant yet t^{d} slightly decreases as shown. It is intuitive that an increase in maximum speed $\bar{v}$ will lead to a decrease in the docking time t^{d}because following AMVs can travel faster to catch up. This suggests that a greater speed limit is appealing to improve docking efficiency without increasing cost when safety is guaranteed in the future. In Fig. 1 (f), as the maximum acceleration $\bar{a}$ increases, $f_{\text{QPM\_SA}}$ increases while $f_{\text{QPM\_UTD}}$, F_{QPM} and t^{d} decrease. The increasing $f_{\text{QPM\_SA}}$ indicates that as the range of acceleration expends, riding comfort during the docking operation becomes worse because bigger acceleration rates can be adopted. This indicates that docking efficiency and mobility can be improved as the acceleration range expands with the cost of sacrificing riding comfort.

Considering the needs of different application scenarios, we also conduct several instances using different weights for the accumulative uncovered travel distance, i.e., c. The results are shown in Table 4. As the weight of the accumulative uncovered travel distance c decreases, i.e., putting less emphasis on mobility performance, the discrepancy between the PH and the QPM can be significant due to the increasing $\mathscr{E}_{\text{SA}}$. However, the absolute value of $\mathscr{E}_{\text{UTD}}$ can still be kept within 0.07 on average, indicating that the PH yields the near-optimal mobility component solution under different application scenarios. The negative value of $\mathscr{E}_{\text{UTD}}$ suggests that the PH can achieve even better mobility performance under some circumstances. However, the increasing $\mathscr{E}_{\text{SA}}$ suggests that the PH should be used with caution as c decreases, and the tradeoff between solution efficiency and solution quality should be more carefully balanced according to specific application needs.

7. Conclusion

This paper proposes a docking (or platooning) trajectory planning problem (DP-TPP) commonly in roadway logistics where long trains/platoons are formed. The DP-TPP docks multiple AMVs (or platoons) with general initial conditions simultaneously while improving mobility, riding comfort, and energy efficiency. Theoretical properties of solution feasibility are analyzed by constructing feasible cones (i.e., local docking feasible cones and global feasible cones) via extending the time geography theory to a second-order version. The theoretical property analysis yields an analytical solution to the minimum docking time. It leads to the development of a parsimonious heuristic trajectory planning algorithm (PH) solving the DP-TPP that achieves near-optimal trajectories specified as a small number of simple quadratic segments. A heuristic alternative solution solving a special case of the DP-TPP is also proposed to obtain the exact optimal trajectories as cubic segments. Then, an exact optimization model (QPM) with a convex objective consisting of riding comfort and mobility performance is proposed. Theoretical properties are used to cut the feasible region of the QPM to improve its solution efficiency.

Numerical experiments are conducted to compare the results of the QPM with and without cuts and the PH. The results show that the PH can achieve a near-optimal solution (with a 5.2% optimality gap on average) yet with a much shorter solution time (0.007 s on average as opposed to the solution time of the QPM). It is appealing for real-time engineering applications that emphasize solution efficiency than solution optimality. It is also noteworthy that the introduction of global feasible cones and optimization starting point save the solution time about 53% on average for those applications that require the exact optimal solutions. Another set of experiments is conducted to compare the PH algorithm with an existing benchmark, i.e., the CACC model. The significant performance gap between the PH solution and the benchmark solution verifies the superiority of optimizing the trajectories in the vehicle docking (or platooning) process with the proposed approach compared with the classic benchmark. According to the sensitivity analysis on the QPM objective, it is better to separate a large AMV fleet into several sub-fleets and dock each sub-fleet at first and then dock sub-fleets together instead of docking all vehicles simultaneously to achieve lower docking costs. Also, a smaller difference between vehicles' average initial speed and docking speed is beneficial in increasing docking efficiency and decreasing costs. A greater maximum speed is appealing to improve docking efficiency in the future when the technology allows. Docking cost and efficiency can be improved at the price of sacrificing riding comfort.

This work provides useful insights into roadway logistics in which platoons/long trains are usually formed, such as transit and

Table 4
The PH and the QPM optimality comparison using different weights.

Weight value	t^d(s)	F_{QPM}	f_{QPM_UTD}	f_{QPM_SA}	F_{PH}	f_{PH_UTD}	f_{PH_SA}	$\mathscr{E}$(%)	$\mathscr{E}_{UTD}$(%)	$\mathscr{E}_{SA}$(%)
$c = 1/5$	14.0	1399.2	1207.1	192.1	1464.7	1290.0	174.7	4.7	6.9	−9.1
$c = 1/10$	14.1	875.9	707.8	168.1	917.7	722.0	195.8	4.8	2.0	16.5
$c = 1/15$	14.3	547.8	404.9	142.8	588.1	416.8	171.3	7.4	2.9	20.0
$c = 1/20$	14.2	508.2	351.0	157.2	551.9	354.0	197.9	8.6	0.9	25.9
$c = 1/50$	13.8	349.5	150.5	199.0	395.5	149.6	245.9	13.2	−0.6	23.6
$c = 1/100$	14.3	209.5	66.5	143.0	251.0	67.0	184.0	19.8	0.8	28.7
$c = 1/200$	14.1	170.8	32.2	138.6	207.6	31.9	175.7	21.5	−0.9	26.8
$c = 1/500$	14.3	168.4	13.8	154.6	213.3	13.8	199.5	26.7	0.0	29.0
$c = 1/800$	13.9	155.1	8.4	146.7	196.8	8.2	188.5	26.9	−2.4	28.5
$c = 1/1000$	14.1	167.3	7.1	160.2	211.1	7.0	204.1	26.2	−1.4	27.4

freight transport. It is suggested to use the proposed docking/platooning operations to improve mobility, riding comfort, and energy efficiency. Further, based on the sensitivity analysis results, logistic managers can select operation parameters properly (e.g., operation speed and the number of vehicles) to balance the tradeoff between operational efficiency and cost.

This work can be extended in several directions in future research. While this study investigates a light traffic condition where there are almost no disruptions from the downstream traffic, it is of great significance to incorporate the impacts of downstream vehicles in denser traffic. This study investigates platooning/docking operations in a pure CAV environment for single-lane traffic. The proposed theoretical properties and solution methods should be modified to suit more complex and realistic environments with multiple lanes and mixed traffic, including human-driven vehicles. Trajectory prediction is necessary if the surrounding vehicle is human-driven/only autonomous without connectivity (Xie et al., 2020). If the surrounding vehicle is connected and autonomous, trajectory prediction is not needed because the intended trajectory can be shared with the platoon via vehicle-to-vehicle technology. Further, the benefits (e.g., fuel efficiency and mobility) yielded from adopting the proposed two-stage docking/platooning operations shall be tested in a large-scale environment, e.g., a corridor. It would be interesting to study vehicle platoon operations at intersections with and without traffic signals where multiple lanes exist and to incorporate lane-changing behaviors. Finally, field experiments can be conducted to test the model feasibility in the real world. Trajectory planning and vehicle control should be properly integrated to ensure safety and guarantee operation performance.

CRediT authorship contribution statement

Qianwen Li: Methodology, Investigation, Writing – original draft. **Xiaopeng Li:** Conceptualization, Methodology, Supervision, Writing – review & editing.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgment

This research is sponsored by National Science Foundation through Grants CMMI #1558887 and CMMI #1932452.

References

Ackerman, E., 2016. nuTonomy to test world's fully autonomous taxi service in Singapore this year. IEEE Spectrum. Available: https://ac. els-cdn. com S.
Alam, A., 2014. Fuel-efficient heavy-duty vehicle platooning.
Alam, A., Gattami, A., Johansson, K.H., Tomlin, C.J., 2014. Guaranteeing safety for heavy duty vehicle platooning: safe set computations and experimental evaluations. Control Eng. Pract. 24, 33–41.
Amoozadeh, M., Deng, H., Chuah, C.-N., Zhang, H.M., Ghosal, D., 2015. Platoon management with cooperative adaptive cruise control enabled by VANET. Veh. Commun. 2, 110–123.
Bang, S., Ahn, S., 2017. Platooning strategy for connected and autonomous vehicles: transition from light traffic. Transp. Res. Rec. 2623, 73–81.
Baskar, L.D., De Schutter, B., Hellendoorn, H., 2008. Dynamic speed limits and on-ramp metering for IVHS using model predictive control. In: in: 2008 11th International IEEE Conference on Intelligent Transportation Systems. IEEE, pp. 821–826.
Bishop, R., Bevly, D., Switkes, J., Park, L., 2014. Results of initial test and evaluation of a driver-assistive truck platooning prototype, in: 2014 IEEE Intelligent Vehicles Symposium Proceedings. IEEE, pp. 208–213.
Bonnet, C., Fritz, H., 2000. Fuel consumption reduction in a platoon: Experimental results with two electronically coupled trucks at close spacing. SAE Technical Paper.
Chen, S., Wang, H., Meng, Q., 2021. An Optimal Dynamic Lane Reversal and Traffic Control Strategy for Autonomous Vehicles. IEEE Trans. Intell. Transp. Syst.
Chen, Z., Li, X., Zhou, X., 2019. Operational design for shuttle systems with modular vehicles under oversaturated traffic: continuous modeling method. Trans. Res. Part B: Methodol. https://doi.org/10.1016/j.trb.2019.05.018.
De Luca, A., Oriolo, G., 2002. Trajectory planning and control for planar robots with passive last joint. Int. J. Robotics Res. 21, 575–590.
Feng, S., Zhang, Y., Li, S.E., Cao, Z., Liu, H.X., Li, L., 2019. String stability for vehicular platoon control: definitions and analysis methods. Annual Rev. Control 47, 81–97.
Firoozi, R., Zhang, X., Borrelli, F., 2021. Formation and reconfiguration of tight multi-lane platoons. Control Eng. Pract. 108, 104714.

Hall, R., Chin, C., 2005. Vehicle sorting for platoon formation: impacts on highway entry and throughput. Trans. Res. Part C: Emerging Technol. 13, 405–420.
Heinovski, J., Dressler, F., 2018. In: Platoon Formation: Optimized Car to Platoon Assignment Strategies and Protocols, in, pp. 1–8.
Horowitz, R., Varaiya, P., 2000. Control design of an automated highway system. Proc. IEEE 88, 913–925.
Jiang, H., Hu, J., An, S., Wang, M., Park, B.B., 2017. Eco approaching at an isolated signalized intersection under partially connected and automated vehicles environment. Trans. Res. Part C: Emerging Technol. 79, 290–307.
Jin, Q., Wu, G., Boriboonsomsin, K., Barth, M., 2013. Platoon-based multi-agent intersection management for connected vehicle, in: 16th International IEEE Conference on Intelligent Transportation Systems (ITSC 2013). IEEE, pp. 1462–1467.
Kunze, R., Ramakers, R., Henning, K., Jeschke, S., 2011. Organization and operation of electronically coupled truck platoons on german motorways, in: Automation, Communication and Cybernetics in Science and Engineering 2009/2010. Springer, pp. 427–439.
Lambert, F., 2017. A new'trackless electric train'(aka a bus) starts testing in China [Online]. Electrek.
Larsen, R., Rich, J., Rasmussen, T.K., 2019. Hub-based truck platooning: potentials and profitability. Trans. Res. Part E: Logistics and Trans. Rev. 127, 249–264. https://doi.org/10.1016/j.tre.2019.05.005.
Larson, J., Liang, K.-Y., Johansson, K.H., 2014. A distributed framework for coordinated heavy-duty vehicle platooning. IEEE Trans. Intell. Transp. Syst. 16, 419–429.
Lee, H., Tomizuka, M., 2003. Adaptive vehicle traction force control for intelligent vehicle highway systems (IVHSs). IEEE Trans. Ind. Electron. 50, 37–47.
Levine, W., Athans, M., 1966. On the optimal error regulation of a string of moving vehicles. IEEE Trans. Autom. Control 11, 355–361.
Li, L., Li, X., 2019. Parsimonious trajectory design of connected automated traffic. Transp. Res. Part B: Methodol. 119, 1–21.
Li, L., Wang, F.-Y., 2007. Advanced motion control and sensing for intelligent vehicles. Springer Sci. & Business Media.
Li, Q., Li, X., Huang, Z., Halkias, J., McHale, G., James, R., 2021. Simulation of mixed traffic with cooperative lane changes. Comput.-Aided Civ. Infrastruct. Eng.
Liang, K.-Y., Mårtensson, J., Johansson, K.H., 2015. Heavy-duty vehicle platoon formation for fuel efficiency. IEEE Trans. Intell. Transp. Syst. 17, 1051–1061.
Liang, K.-Y., Mårtensson, J., Johansson, K.H., 2013. When is it fuel efficient for a heavy duty vehicle to catch up with a platoon? IFAC Proceedings Vol. 46, 738–743.
Litman, T., 2003. Measuring transportation. Traffic, mobility and accessibility. ITE J. 73, 28–32.
Liu, H., Kan, X.D., Shladover, S.E., Lu, X.-Y., Ferlis, R.E., 2018. Modeling impacts of cooperative adaptive cruise control on mixed traffic flow in multi-lane freeway facilities. Trans. Res. Part C: Emerging Technol. 95, 261–279.
Liu, P., Huda, M.N., Sun, L., Yu, H., 2020. A survey on underactuated robotic systems: bio-inspiration, trajectory planning and control. Mechatronics 72, 102443.
Liu, X., Qu, X., Ma, X., 2021. Improving flex-route transit services with modular autonomous vehicles. Trans. Res. Part E: Logistics and Trans. Rev. 149, 102331 https://doi.org/10.1016/j.tre.2021.102331.
Lu, X.-Y., Shladover, S.E., 2014. Automated truck platoon control and field test. Road Vehicle Automation. Springer 247–261.
Ma, J., Zhou, F., Demetsky, M.J., 2012. Evaluating mobility and sustainability benefits of cooperative adaptive cruise control using agent-based modeling approach. In: in: 2012 IEEE Systems and Information Engineering Design Symposium. IEEE, pp. 74–78.
Maiti, S., Winter, S., Kulik, L., 2017. A conceptualization of vehicle platoons and platoon operations. Trans. Res. Part C: Emerging Technol. 80, 1–19.
Malikopoulos, A.A., Cassandras, C.G., Zhang, Y.J., 2018. A decentralized energy-optimal control framework for connected automated vehicles at signal-free intersections. Automatica 93, 244–256.
Melzer, S.M., Kuo, B.C., 1971. Optimal regulation of systems described by a countably infinite number of objects. Automatica 7, 359–366.
Milanés, V., Godoy, J., Villagrá, J., Pérez, J., 2010. Automated on-ramp merging system for congested traffic situations. IEEE Trans. Intell. Transp. Syst. 12, 500–508.
Milanés, V., Shladover, S.E., 2014. Modeling cooperative and autonomous adaptive cruise control dynamic responses using experimental data. Trans. Res. Part C: Emerging Technol. 48, 285–300.
Morales, A., Nijmeijer, H., 2016. Merging strategy for vehicles by applying cooperative tracking control. IEEE Trans. Intell. Transp. Syst. 17, 3423–3433.
Noruzoliaee, M., Zou, B., Zhou, Y.(J.), 2021. Truck platooning in the U.S. national road network: a system-level modeling approach. Trans. Res. Part E: Logistics and Trans. Rev. 145, 102200 https://doi.org/10.1016/j.tre.2020.102200.
Ploeg, J., Shukla, D.P., van de Wouw, N., Nijmeijer, H., 2013. Controller synthesis for string stability of vehicle platoons. IEEE Trans. Intell. Transp. Syst. 15, 854–865.
Stankovic, S.S., Stanojevic, M.J., Siljak, D.D., 2000. Decentralized overlapping control of a platoon of vehicles. IEEE Trans. Control Syst. Technol. 8, 816–832.
Tan, H.-S., Rajamani, R., Zhang, W.-B., 1998. Demonstration of an automated highway platoon system, in: Proceedings of the 1998 American Control Conference. ACC (IEEE Cat. No. 98CH36207). IEEE, pp. 1823–1827.
Tank, T., Linnartz, J.-P., 1997. Vehicle-to-vehicle communications for AVCS platooning. IEEE Trans. Veh. Technol. 46, 528–536.
Tarek, F., 2018. Dubai tests autonomous pods in drive for smart city [Online]. Reuters.
Tsugawa, S., 2013. An overview on an automated truck platoon within the energy ITS project. IFAC Proceedings Volumes 46, 41–46.
Turri, V., Kim, Y., Guanetti, J., Johansson, K.H., Borrelli, F., 2017. A model predictive controller for non-cooperative eco-platooning. In: in: 2017 American Control Conference (ACC). IEEE, pp. 2309–2314.
Valente, A.S., Montanaro, U., Tufo, M., Salvi, A., Santini, S., 2014. Design of a platoon management strategy and its hardware-in-the loop validation. In: in: 2014 IEEE 79th Vehicular Technology Conference (VTC Spring). IEEE, pp. 1–5.
Van De Hoef, S., Johansson, K.H., Dimarogonas, D.V., 2015. Fuel-optimal centralized coordination of truck platooning based on shortest paths. In: in: 2015 American Control Conference (ACC). IEEE, pp. 3740–3745.
Wang, Z., Zhao, X., Xu, Z., Li, X., Qu, X., 2020. Modeling and field experiments on lane changing of an autonomous vehicle in mixed traffic. computer-aided civil and infrastructure. Engineering.
Wei, Y., Avcı, C., Liu, J., Belezamo, B., Aydın, N., Li, P.T., Zhou, X., 2017. Dynamic programming-based multi-vehicle longitudinal trajectory optimization with simplified car following models. Trans. Res. part B: methodol. 106, 102–129.
Wei, Y., Liu, J., Li, P., Zhou, X., 2016. Longitude trajectory optimization for autonomous vehicles: an approach based on simplified car-following model. in: Transportation Research Board 95th Annual Meeting.
Xie, G., Shangguan, A., Fei, R., Ji, W., 2020. Motion trajectory prediction based on a CNN-LSTM sequential model 63, 1–21.
Xu, Z., Wang, Y., Wang, G., Li, X., Yuan, Q., Zhao, X., 2019. Trajectory optimization for a connected automated traffic stream. Comparison Between An Exact Model and Fast Heuristics.
Yao, H., Cui, J., Li, X., Wang, Y., An, S., 2018. A trajectory smoothing method at signalized intersection based on individualized variable speed limits with location optimization. Trans. Res. Part D: Trans. Environ. 62, 456–473.
Zhang, W., Jenelius, E., Ma, X., 2017. Freight transport platoon coordination and departure time scheduling under travel time uncertainty. Trans. Res. Part E: Logistics and Trans. Rev. 98, 1–23. https://doi.org/10.1016/j.tre.2016.11.008.
Zhang, Y., Malikopoulos, A.A., Cassandras, C.G., 2017. Decentralized optimal control for connected automated vehicles at intersections including left and right turns. In: in: 2017 IEEE 56th Annual Conference on Decision and Control (CDC). IEEE, pp. 4428–4433.
Zhuang, W., Xu, L., Yin, G., 2020. Robust cooperative control of multiple autonomous vehicles for platoon formation considering parameter uncertainties. Automotive Innovation 1–13.